Ministerul Educaţiei, Culturii și Cercetării al Republicii Moldova

Universitatea Tehnică a Moldovei

Departamentul Informatică și Ingineria Sistemelor

**RAPORT**

Lucrare de laborator Nr.6

Disciplina: APSI

Tema: Dezvoltarea elaborărilor prin intermediul diagramelor de stare și diagramelor de activităţi.

A efectuat: st.gr.

Ulmanu Cristian IA-182,

A verificat : lect.univ.   
 Sava Nina  
Melnic Radu

Chișinău 2020

**Scopul:** studierea noțiunilor de stare, stare compusă cu substari concurente, depuse și disjuncte, activitate, condiție de gardă, bară de sincronizare (fork, join), bloc de decizie.

**Sarcină:** de realizat 3 diagrame de stare și 3 diagrame de activitate pentru sistemul informațional ales

**Tema:** sistem de operare.

**Considerații teoretice**

Diagramele de stare au la bază două concepte importante: stările ce caracterizează valori sau seturi de valori ale obiectelor, și evenimete ce constau din stimuliexterni, ce acționează asupra obiectelor și determină o tranziție de la o stare la alta. Diagramele de stare sunt grafuri, în care nodurile sunt stări și arcele direcționate sunt tranziții, avînd ca etichete numele evenimentelor care le-au provocat. Se precizează acțiunile rezultate din aceste schimbări.

Diagramele de stare sunt create numai pentru clasele cu un comportament dinamic semnificativ,care se pot determine prin studierea diagramelor de interacțiune, determinate la rîndul lor din descrierea diagramelor cazurilor de utilizare. Astfel, comportamentul dinamic al sistemului este caracterizt printr-o mulțime de obiecte concurente, fiecare cu propria diagramă de stare.

De fapt, conceptul de diaramă de stare corespunde în literatura tehnică mașinilor cu stări finite, care sunt tratate în moduri diferite, în funcție de domeniul de utilizare.

Starea corespunde unui interval de timp în care obiectul satisface anumite condiții, efectuiază anumite acțiuni sau așteaptă un eveniment. Starea se reprezintă în UML printr-un dreptunghi cu colțurile rotungite, ce poate avea un compartiment pentru nume și altul pentru tranzițiile interne, care pot fi acțiuni sau activități efectuate în timp ce elementul se află în acea stare. Acțiunile sunt precedate de o etichetă urmată de caracterul '/' și, chiar dacă limbajul UML nu are o termenologie pentru a face diferențierea între acțiunile instantanee și cele care durează un interval de timp, această caracteristicătrebuie identificată la analiza dinamică a obiectelor.

Exemple de etichete des utilizate:

* entry - pentru o acțiune considerată instantanee, efectuată la intrarea în stare;
* do - pentru o activitate ce durează un interval de timp, fie pînă ce este finalizată, fie pînă la ieșirea din starea corepunzătoare;
* tranzițieInternă - pentru a reprezenta o acțiune instatanee, executată la producerea unui eveniment care nu determină tranziția în altă stare; aceasta seamănă cu o tranziție de la o stare către ea însăși cum este NumeEveniment2 din figura 2, dar fără ieșire din stare și reintrare, deci fără executarea acțiunilor specificate prin exit și entry;
* exit - pentru o acțiune considerată instantanee, executată la eșirea din stare.

Diagramele de activități sunt cazuri particulare ale diagramelor de stare care nu descriu însă un flux de control bazat pe evenimente, ci unul procedural, în care toate sau majoriatea tranzițiilor se efectuează automat, la terminarea acțiunilor efectuate în interiorul stărilor. Diagramele de activități se pot reprezenta pentru a descrie un caz de utilizare, un pachet sau implementarea unei operații.

În diagramele de activitatepot apărea două tipuri particulare de stări: stări acțiune și stări subactivitate. O stare acțiune, spre deosebire de stările obișnuite, nu poate avea tranziții internesau tranziții de ieșire bazate pe evenimente; ea reprezintă o etapă din execuția unui algoritm sau a unui flux de lucru, avînd asociată o acțiune de intrare, la teminarea căreia se generează automat o tranziție de iețire din stare.

**Diagramele de stare pentru sistemul de operare**

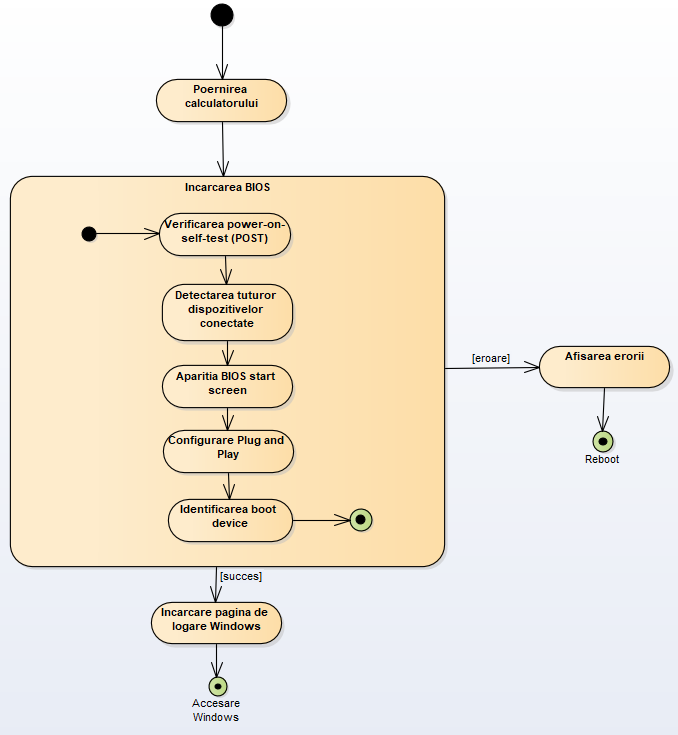


Figura 1: Diagrama de stare care reprezinta procesul de booting a Windows

Această diagramă redă stările și tranzițiile care au loc în cadrul procesului de booting a unui sistem de operare. Începând cu Pornirea calculatorului și continuând cu Incarcare BIOS care are mai multe stări care merg consecutiv una după alta, dacă procesul de încărcare a mers corect atunci apare fesreastra de loghare, iar în caz de eroare este necesar reboot.

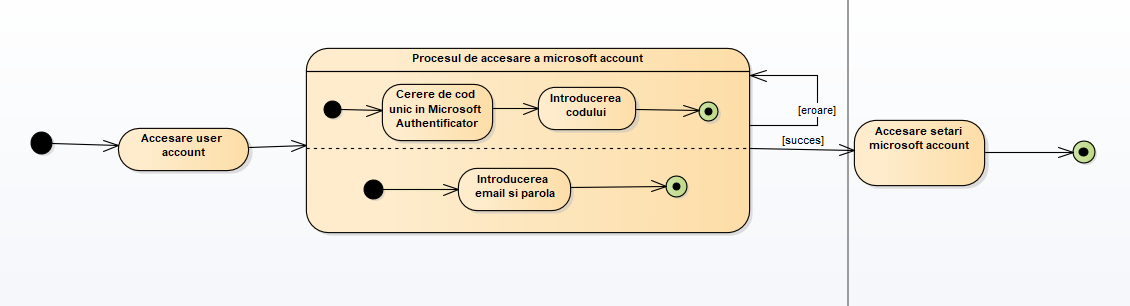


Figura 2: Diagrama de stare care reprezintă procesul de loghare la Microsoft account

Diagrama dată arată procesul de accesare la microsoft account, în cadrul diagramei noi observăm ca aici câteva stări merg paralel și anume introducerea parolei și generarea unui cod unic în cadrul aplicației Microsoft Authentificator necesar pentru logharea în account.

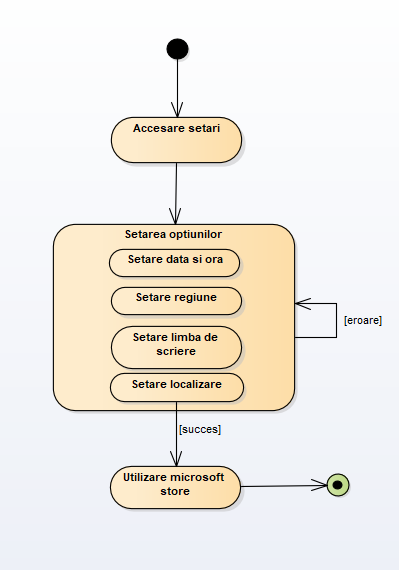


Figura 3: Diagrama de stare pentru setarea opțiunilor Windows

Diagrama dată arată care sunt stările pentru setarea opțiunilor necesare pentru utilizarea microsoft store din Windows. Stările din setarea opțiunilor nu au o ordine cronologică fixă, astfel fiecare setare poate fi făcută în orice ordine.

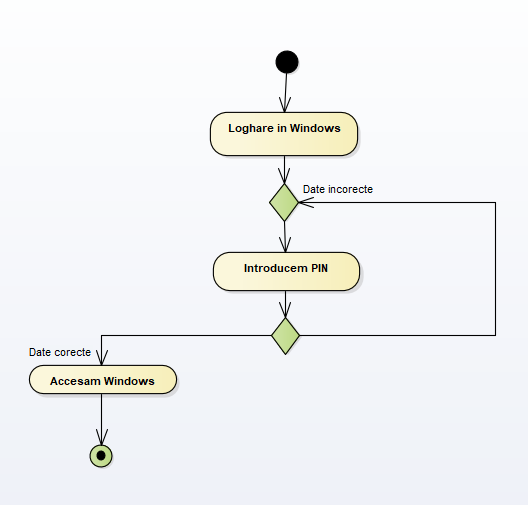


Figura 4: Diagrama de activități pentru logharea în Windows

În diagrama sunt reprezentate activitățile necesare pentru logharea în Windows. Avem o ramificare și un bloc de decizie pentru verificarea dacă datele la loghare au fost introduse corect.

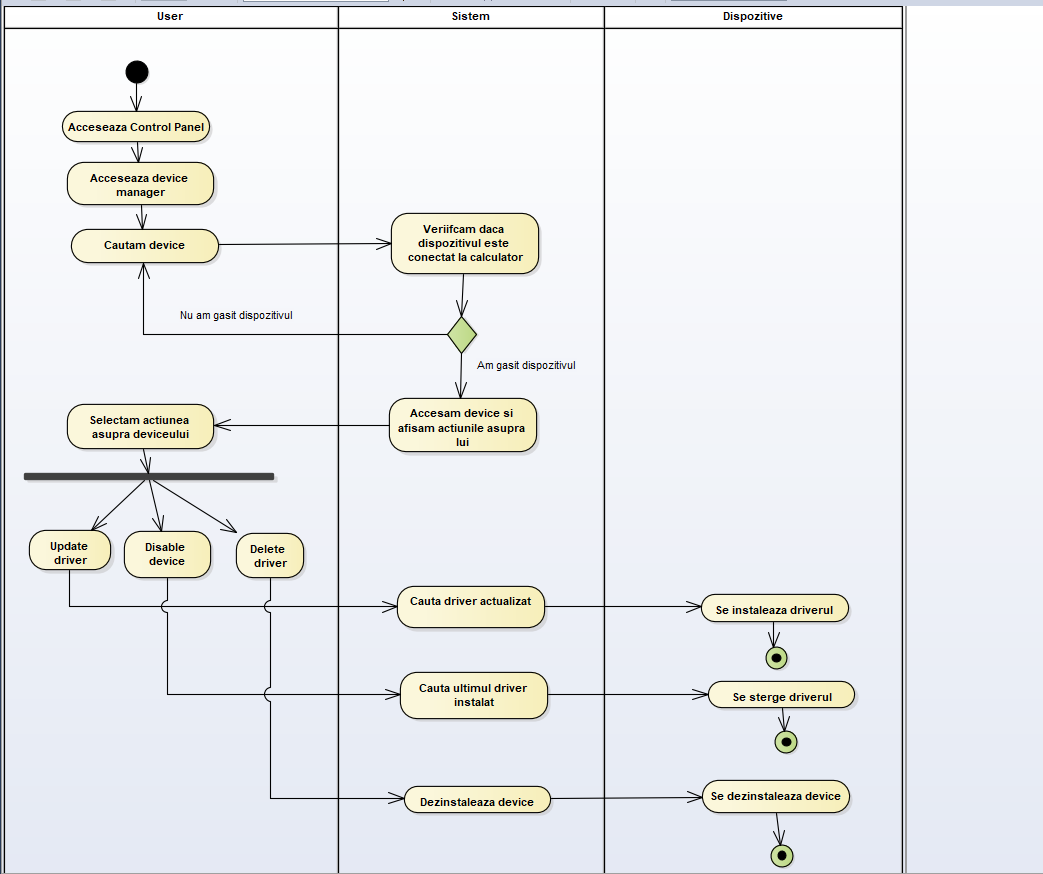


Figura 5: Diagrama de activități pentru device manager din Windows

În diagrama dată sunt 3 partiții User, Sistem și Dispozitive, aici este descris procesul de administrare a unui dispozitiv în device manager. Avem o ramificare care verifică dacă este dispozitivul conectat sau nu, avem fork unde putem alege care va fi acțiunea asupra deviceului.

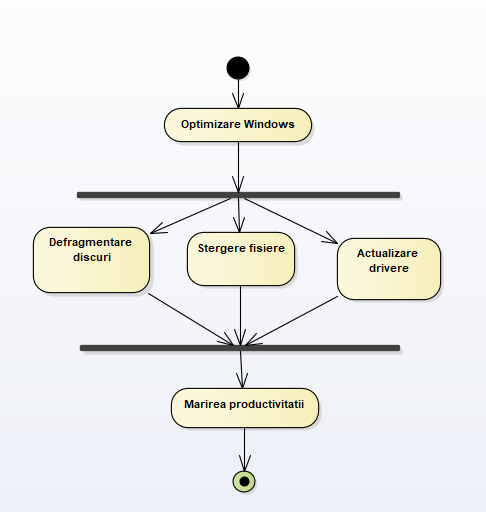


Figura 6: Diagrama de activități pentru optimizare Windows

În diagrama dată sunt arătate activitățile necesare pentru optimizarea Windows. Este arătat un fork cu activitățile pentru optimizare care se unesc într-un join care are ca rezultat mărirea productivității.

**Concluzii**

În cadrul aceste lucrări de laborator eu am făcut cunoștință cu conceptul de diagramă de clase din limbajul UML. Asemenea diagrame permit reprezentarea stărilor, tranzițiilor sau activităților din cadrul unui sistem. Față de digrama de stare, cea de activități ne permite să adăugăm și ramificare, arbore de decizie, pentru a demonstra care este consecutivitatea acitivităților în cazul unei sau altei opțiuni. De asemenea încă o diferență sunt forkurile și joinurile care permit separarea și unirea activităților. Partițiile ne permit să reprezentăm de cine sunt effectuate anumite activități și să facem mai detaliate diagramele.

**Bibliografie**

1. **Melnic R., Sava N.** Indrumar metodic “Analiza si modelarea sistemelor informationale”.
2. Teoria de la cursul APSI.
3. Diagrame de stare [Reusrsă electronică]

<https://sites.google.com/site/uml4students/diagrama-de-stare>

1. Diagrame de activitate [Reusrsă electronică]

https://sites.google.com/site/uml4students/diagrama-de-activitate