Ministerul Educaţiei, Culturii și Cercetării al Republicii Moldova

Universitatea Tehnică a Moldovei

Departamentul Ingineria Software și Automatică

**RAPORT**

Lucrare de laborator Nr.7

Disciplina: Analiza Modelarea Sistemelor

Tema: Dezvoltarea elaborărilor prin intermediul diagramelor de componente şi de plasare

A efectuat: st.gr. Vlașițchi Ștefan TI-216,

A verificat : asist.univ.   
 Sava Nina

lect.univ.  
Melnic Radu

Chișinău 2023

**Scopul**: Studierea noțiunilor de component, interfața, nod, dependență, conexiune între noduri.

**Sarcina:** de realizat 3 diagrame de componente și 2 diagrame de plasare pentru sistemul informațional ales.

**Considerații teoretice**

O diagramă de stare este un instrument utilizat pentru a descrie comportamentul unui sistem din punctul de vedere al stărilor pe care le poate avea și a tranzițiilor între acestea. Ea este formată din stări, tranziții și evenimente care declanșează tranzițiile.

Diagrama începe cu starea inițială, reprezentată de un punct, și apoi definește toate stările posibile și tranzițiile între acestea. În general, o stare este reprezentată printr-un pătrat cu numele stării, iar tranzițiile sunt reprezentate printr-o săgeată care pornește de la o stare și se termină în alta. Săgeata este etichetată cu evenimentul care declanșează tranziția.

În cazul unui sistem complex, diagrama de stare poate fi foarte mare și poate conține sub-stări și tranziții. De asemenea, pot fi definite acțiuni care se întâmplă în cadrul unei stări sau tranziții, cum ar fi apelul unei funcții sau modificarea unei variabile.

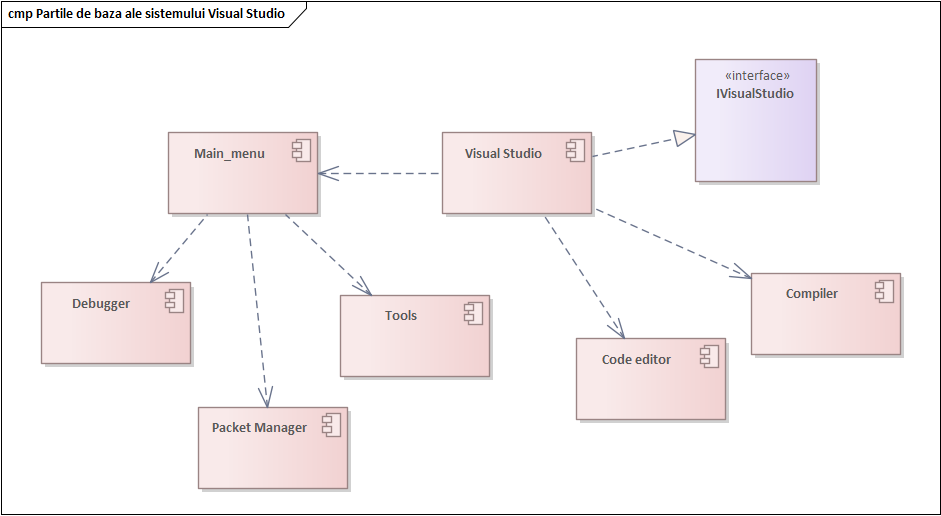
Diagrama de stare este utilă pentru a înțelege comportamentul unui sistem și pentru a identifica eventuale probleme sau incoerențe în comportamentul acestuia. Ea poate fi utilizată în procesul de dezvoltare software pentru a ajuta la proiectarea și implementarea sistemului, dar și pentru a testa și depana aplicația.

O diagramă de activitate este o reprezentare grafică a fluxului de lucru, arătând activitățile și acțiunile care sunt efectuate și deciziile luate în timpul procesului. Aceasta este formată din forme geometrice, cum ar fi ovalul, rombul, dreptunghiul și săgeți, care reprezintă acțiuni, decizii, bifurcări, îmbinări și fluxuri de date. Diagrama de activitate poate fi utilizată pentru a modela diverse procese de afaceri, fluxuri de lucru și algoritmi.

Într-o diagramă de activitate, activitățile sunt reprezentate de obicei de dreptunghiuri, iar săgețile indică direcția fluxului de lucru. Deciziile sunt reprezentate de obicei de romburi și pot fi utilizate pentru a modela bifurcări în proces, în funcție de anumite condiții. Bifurcările și îmbinările pot fi reprezentate de obicei de diamante și indică fluxurile alternative și fuziunea lor. Fluxurile de date pot fi reprezentate prin săgeți cu linie punctată, care arată transmiterea datelor între activități.

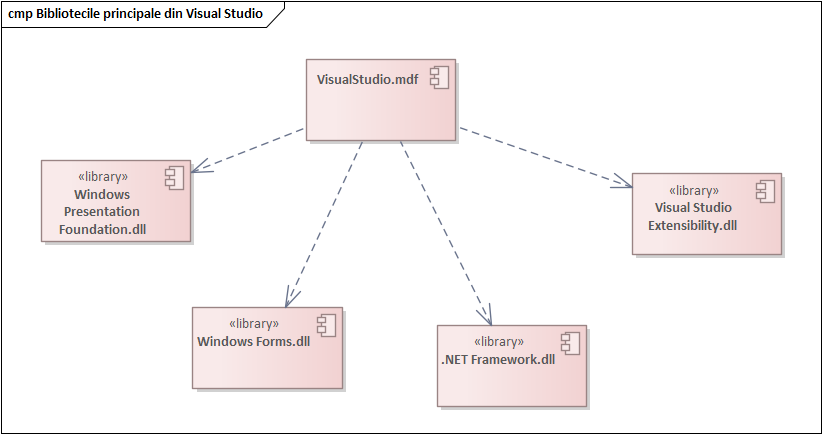
Diagrama de activitate poate fi folosită pentru a modela fluxurile de lucru și procesele de afaceri, pentru a identifica și rezolva problemele într-un proces, pentru a îmbunătăți eficiența procesului și pentru a îmbunătăți comunicarea între departamente și echipe. Este important să se acorde atenție detaliilor și să se asigure că diagrama de activitate este ușor de înțeles și de utilizat de către toți cei implicați în procesul modelat.

**Implementare, rezultate practice:**



***Figura 1.*** Părțile de baza ale sistemului Visual Studio

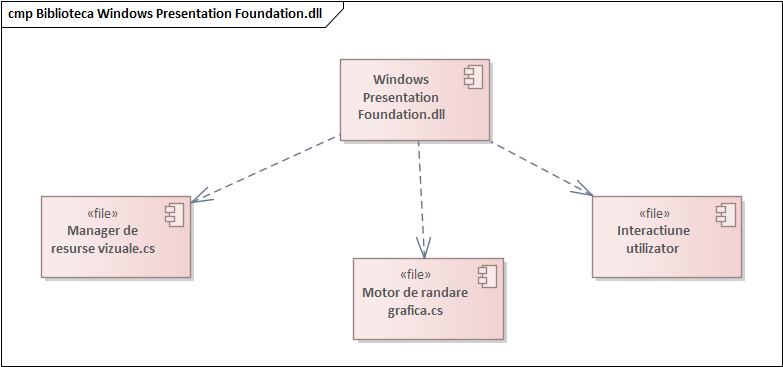
În figura 1, este reprezentată o diagramă de componente în care se arată părțile de bază ale sistemului Visual Studio. Una dintre părțile principale este meniul, în care sunt accesibile toate funcțiile oferite de sistem. Pe lângă acesta, mai avem și alte părți principale, precum editorul de cod și compilatorul, iar în cele din urmă, avem și interfața utilizatorului.

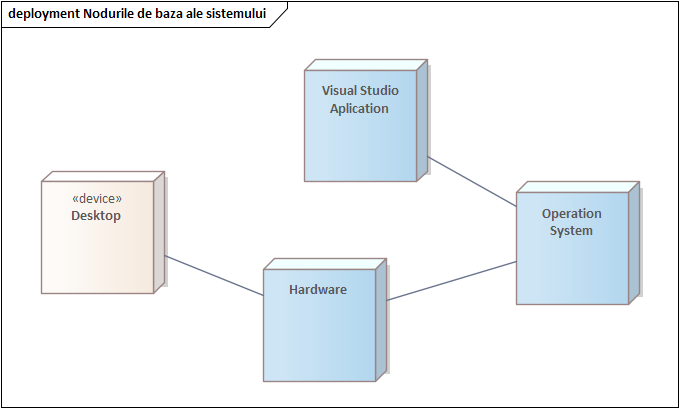
***Figura 2.***Bibliotecile principale din Visual Studio

Aplicația Visual Studio folosește o serie de biblioteci și componente interne pentru a oferi funcționalități și servicii diverse. În cele ce urmează, voi enumera câteva dintre bibliotecile principale utilizate de aplicația Visual Studio:

* Windows Presentation Foundation (WPF): Este o bibliotecă pentru dezvoltarea interfețelor grafice (GUI) în aplicații Windows. Visual Studio folosește WPF pentru a crea interfața utilizator.
* Windows Forms (WinForms): Este o bibliotecă mai veche pentru crearea de interfețe grafice în aplicații Windows. Deși este înlocuită în mare măsură de WPF, unele componente mai vechi din Visual Studio pot folosi încă Windows Forms.
* .NET Framework: Este o platformă de dezvoltare a aplicațiilor pentru sistemele Windows. Visual Studio se bazează pe .NET Framework pentru a furniza funcționalități extinse și pentru a dezvolta aplicații Windows.
* Visual Studio Extensibility (VSX): Este o bibliotecă care permite dezvoltatorilor să creeze extensii și adăugări personalizate pentru Visual Studio. Acesta oferă seturi de instrumente și API-uri pentru a extinde funcționalitatea IDE-ului.
* Visual Studio Shell: Este o componentă care oferă infrastructura de bază pentru construirea aplicației Visual Studio. Aceasta include instrumentele necesare pentru crearea de ferestre, meniuri, bare de instrumente și alte elemente de interfață.

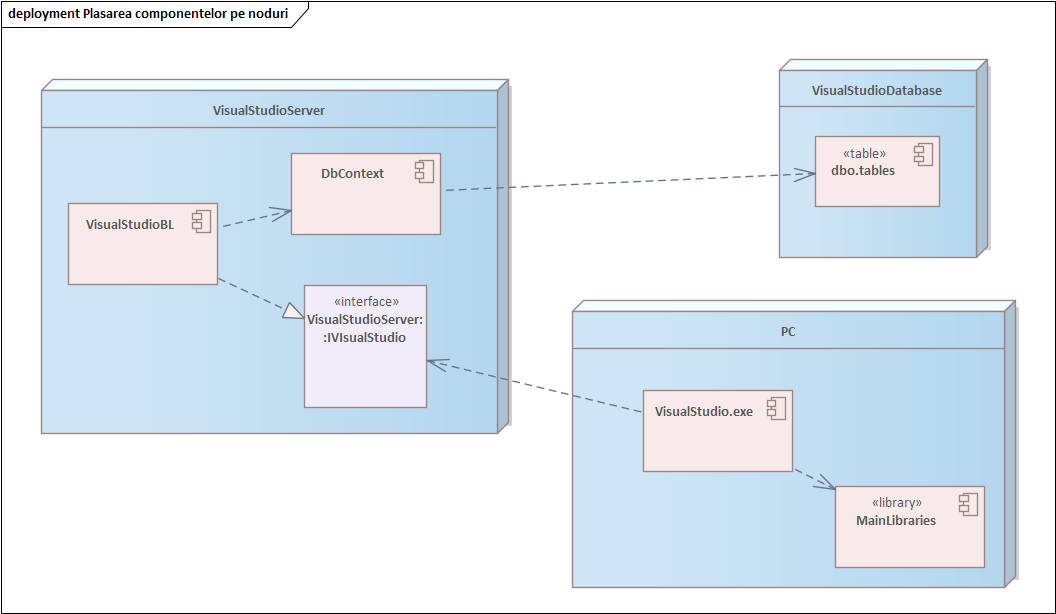
Acestea sunt doar câteva dintre bibliotecile și componentele principale utilizate de aplicația Visual Studio. Există multe alte biblioteci și componente interne care sunt utilizate în cadrul IDE-ului pentru a oferi diverse funcționalități, cum ar fi analiza codului, depanarea, gestionarea proiectelor, controlul versiunilor și multe altele.

 ***Figura 3.*** Biblioteca Windows Presentation Foundation

In figura 3 sunt reprezentate fisierele principale pentru biblioteca Windows Presentation Foundation

***Figura 4.*** Nodurile de baza ale sistemului

In figura 4 este reprezentat nodul de baza prin care aplicatie visual

***Figura 5.*** Plasarea componentelor pe noduri

In figura 5 este prezentata relatia dintre componentele ale sistemului VisualStudio plasate pe noduri diferite. Pe PC se afla fisierul executabil care cand este rulata este insotit de fisierele principale pentru ai permite functionarea in regimul online.Serverul detine logica da baza a sistemului plus Contextul sistemului care are o relati ede dependenta cu tabelele din baza de date.

**Concluzii:**

Dezvoltarea elaborărilor prin intermediul diagramelor de componente și de plasare aduce claritate și coeziune în procesul de dezvoltare software. Aceste diagrame oferă o imagine detaliată asupra structurii și distribuției aplicației, facilitând înțelegerea și comunicarea între membrii echipei de dezvoltare. Prin evidențierea relațiilor dintre componente și infrastructura fizică, dezvoltatorii pot lua decizii informate în ceea ce privește arhitectura și implementarea sistemului. Utilizarea acestor diagrame îmbunătățește calitatea, eficiența și gestionarea proiectelor software, conducând la dezvoltarea cu succes a aplicațiilor Visual Studio și a altor sisteme complexe.

**Bibliografie**

1. **Melnic R., Sava N.** Indrumar metodic “Analiza si modelarea sistemelor informationale”.
2. **Introduction to Visual Studio :** <https://www.geeksforgeeks.org/introduction-to-visual-studio/>
3. **Aplicatia GIT:** <https://github.com/>
4. **Crearea unei diagrame de plasare UML** : <https://cadredidactice.ub.ro/simonavarlan/files/2018/10/L7_diagrame_de_implementare.pdf>
5. **Install NuGet client tools** : <https://learn.microsoft.com/en-us/nuget/install-nuget-client-tools>
6. **Creați o diagramă de componente UML:** <https://support.microsoft.com/es-es/office/crear-un-diagrama-de-componentes-uml-aa924ecb-e4d2-4172-976e-a78fa157b074>