

[Titlul temei]

[data prezentării]

Îndrumător:

dr. ing. Daniel Morariu

Student:

**Nume, Prenume
(grupa/semigrupa)**

Istoric Versiuni

Data	Versiune	Descriere	Autor
<dd/mm/yy>	<x.x>	<detalii>	<nume>

[În această secțiune puteți păstra informații despre când ați rezolvat o anumită problemă din proiect. Aveți astfel o evoluție a implementării proiectului realizat. Eventual puteți scrie în coloana de autor și cine v-a ajutat în cazul în care nu ați rezolvat o anumită problemă singuri. Pot să existe în proiect anumite părți mici făcute de alt cineva în cazul în care nu ați înțeles momentan cum se rezolvă problema respectivă și ați cerut ajutor de la un coleg/ prieten.

Atenție în continuare și la aranjarea documentului în pagină, contează și acest lucru.

Textele scrise cu albastru în document conțin doar informații despre ce trebuie scris la secțiunea respectivă, ulterior pot fi șterse din document.

Titlul proiectului poate să fie numele dat de d-voastră pentru proiectul realizat (nu trebuie să fie numele din proiectul din lista de proiecte).]

Cuprins

ISTORIC VERSIUNI	2
CUPRINS	3
1 SPECIFICAREA CERINTELOR SOFTWARE	4
1.1 Introducere	4
1.1.1 Obiective	4
1.1.2 Definiții, Acronime și Abrevieri	4
1.1.3 Tehnologiile utilizate	4
1.2 Cerințe specifice.....	4
2 FUNCȚIONALITATE.....	5
2.1 Descriere.....	5
2.2 Fluxul de evenimente	5
2.2.1 Fluxul de bază	5
2.2.2 Fluxuri alternative	5
2.2.3 Pre-condiții.....	5
2.2.4 Post-condiții	5
3 IMPLEMENTARE	6
3.1 Diagrama de clase.....	6
3.2 Descriere detaliată.....	7
4 BIBLIOGRAFIE	8

1 Specificarea cerințelor software

1.1 Introducere

[Această secțiune va conține o descriere pe scurt a ceea ce trebuie să facă proiectul, cerințele proiectului. Imagine de ansamblu a ceea ce conține proiectul.]

1.1.1 Obiective

[În această secțiune se vor specifica obiectivele propuse spre a fi realizate pentru acest proiect. Se va specifica atât ceea ce s-a realizat precum și cerințele care sunt nefuncționale datorită constrângerilor de proiectare sau datorită altor factori (nu a fost suficient timp pentru a finaliza proiectul). Aici se va prezenta tot ceea ce considerați că ar trebui realizat pentru ca acest proiect să îl considerați că este complet.]

1.1.2 Definiții, Acronime și Abrevieri

[În această secțiune se poate furniza definițiile tuturor termenilor, acronimelor sau abrevierilor care au fost folosite în dezvoltarea aplicației software sau în scrierea documentației. De asemenea dacă sunt folosiți termeni mai speciali aici se poate scrie definiția sau ce înseamnă pentru ca să fie de ajutor în timpul prezentării proiectului.]

1.1.3 Tehnologiile utilizate

[În această secțiune se vor menționa toate tehnologiile utilizate pentru dezvoltarea proiectului (atât a aplicației software cât și a documentației aferente). Dacă s-au folosit obiecte speciale (de exemplu clase/biblioteci create de altcineva pentru a folosi butoane mai speciale în c++ Builder sau alte clase/biblioteci care nu sunt incluse în pachetul standard al mediului în care s-a dezvoltat aplicația) vor fi prezentate în această secțiune.]

1.2 Cerințe specifice

[Această secțiune ar trebui să conțină doar obiectivele propuse în secțiunea 1.1.1 și care au fost și realizate efectiv. Aceste obiective o să le numim în continuare funcționalități ale aplicației realizate. În această secțiune se vor enumera toate funcționalitățile implementate ale aplicației și eventual o scurtă descriere a ceea ce face fiecare funcționalitate în parte.

Urmează ca în secțiunea următoare (secțiunea 2) să se prezinte detaliat cel puțin două dintre aceste funcționalități (se va realiza câte un capitol 2 separat pentru fiecare funcționalitate descrisă separat). Se vor prezentat funcționalitățile care sunt considerate ca fiind cele mai relevante pentru aplicația realizată].

Exemplu de funcționalități:

- afișarea pieselor pe tabla de joc
- mutarea pieselor pe tablă
- rotirea tablei de joc când este afișată la un jucător sau la alt jucător (în cazul în care este nevoie)
- codificarea realizată pentru mesajele transmise / recepționate pe rețea
- etc.

2 Funcționalitate

2.1 Descriere

[Aici se va specifica de ce se consideră că funcționalitatea aleasă pentru a fi prezentată este relevantă în aplicația curentă. Această secțiune cuprinde o scurtă descriere a funcționalității care va fi prezentată.]

2.2 Fluxul de evenimente

2.2.1 Fluxul de bază

[În această secțiune se va detalia modul de realizare a funcționalității prezentate în această secțiune. Se vor descrie în pași detaliați modul de implementare a funcționalității respective. Această descriere trebuie să conțină tot ce trebuie să facă utilizatorul pentru a porni această funcționalitate, ce face sistemul ca răspuns la acea funcționalitate și ce ar trebui să obțină utilizatorul la ieșire.

Dacă se consideră necesar în această secțiune se pot prezenta și porțiuni de cod care se consideră de autor a fi relevant și care realizează o parte din funcționalitate. Fiecare cod trebuie însoțit de comentarii.

Dacă funcționalitatea descrisă a fost apelată din mai multe locuri se va detalia toate locurile din care a fost apelată și modul în care a fost apelată.

Dacă aceeași funcționalitate a fost implementată în mai multe locuri în aplicație iar în fiecare loc a fost implementată altfel atunci se va descrie în secțiunea Fluxuri alternative fiecare mod de implementare în parte. Dacă funcționalitatea a fost implementată doar într-un singur loc se va renunța la secțiunea 2.2.2.]

2.2.2 Fluxuri alternative

2.2.2.1 < Primul flux alternativ >

[Se va detalia implementarea făcută în cazul în care funcționalitatea curentă a fost rezolvată și altfel decât rezolvarea de la fluxul de bază, pe motiv că ulterior s-au găsit și alte moduri mai optime de a implementa un anumit lucru. Fiecare mod de implementare va ocupa o secțiune separată.]

2.2.2.2 < Al doilea flux alternativ >

[Un alt mod folosit pentru a implementa funcționalitatea curentă.]

2.2.3 Pre-condiții

[Această secțiune conține o listă cu toate acțiunile care trebuie realizate de către utilizator pentru a porni în mod corect funcționalitatea curentă. Conține acțiunile și setările care trebuie realizate pentru o bună funcționare a aplicației (funcționalității descrise). Se vor detalia toate problemele care nu au fost momentan rezolvate și pot aduce aplicația într-o stare nefuncțională. De exemplu pe ce butoane trebuie apăsat și în ce ordine, în cazul în care la început toate butoanele sunt active.]

2.2.4 Post-condiții

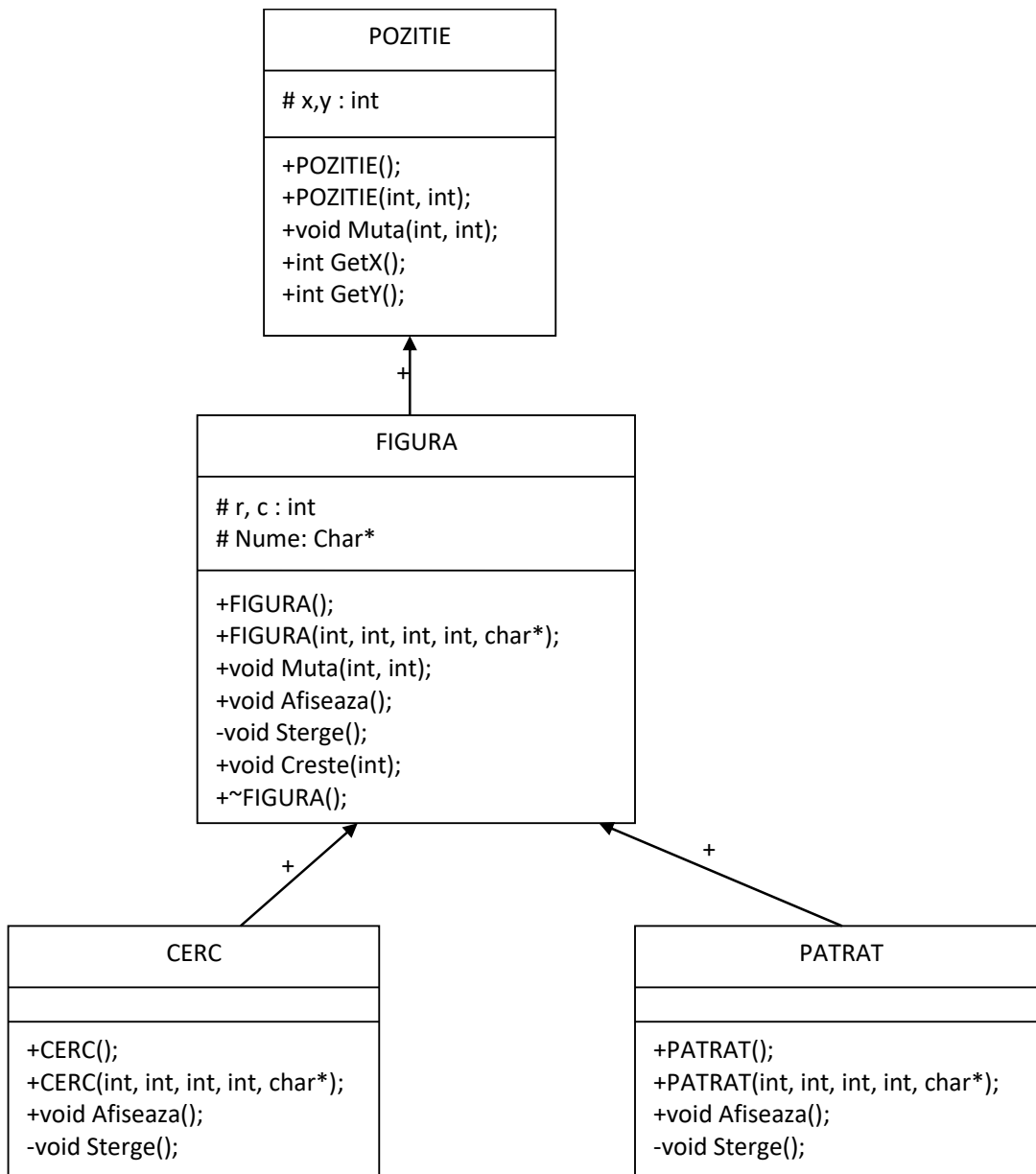
[Această secțiune conține o listă cu ceea ce obține utilizatorul după rularea funcționalității prezentate. La ce ar trebui să se aștepte utilizatorul după rularea funcționalității, ce vede el pe ecran, ce ar trebui ca să facă ca să vadă că funcționalitatea s-a terminat și că s-a terminat corect.]

3 Implementare

3.1 Diagrama de clase

[Această secțiune va include o imagine cu diagrama claselor dezvoltate în aplicație (în special clasele proprii). Diagrama va conține numele clasei, membrii și metodele precum și relațiile dintre clase. Se recomandă folosirea unor notații standard (a se vedea informațiile de la adresa <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/uml-class-diagram-tutorial/>). Dacă se folosesc alte notații acestea se vor descrie în secțiunea 1.1.2. Ca și exemplu diagrama de clase pentru clasele de la laborator.

Pentru secțiunea 3.1 și 3.2 în cazul în care nu se reușește desenarea pe calculator accept și scheme făcute frumos de mână]

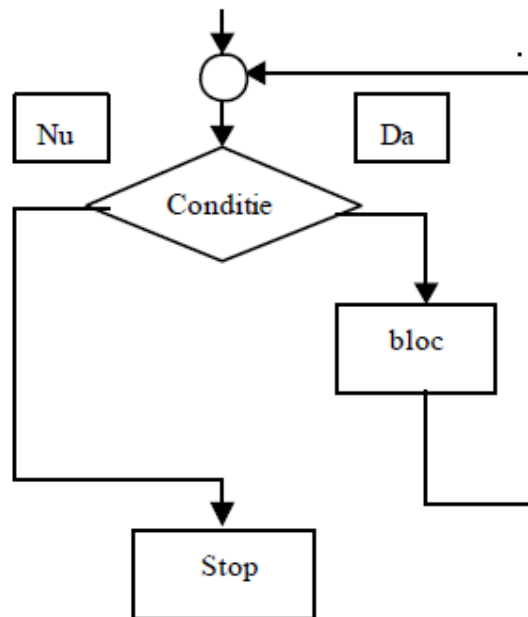


3.2 Descriere detaliată

[Această secțiune va conține o schemă logică de funcționare a aplicației. Schema va conține toate blocurile funcționale considerate importante pentru aplicația curentă. Nu trebuie făcută schema detaliată a întregii aplicații ci, acum după implementare, cum ați considera că funcționează aplicația realizată]

Exemplu schemă logică:

Schema logica



4 Bibliografie

[Se vor specifica sursele bibliografice folosite pentru a realiza acest proiect. Cărțile și site-urile web folosite pentru partea de Programare orientată obiect și partea de programare în C++ Builder. Pentru site-urile web se va specifica numele paginii respective și adresa de web. Mai jos 2 exemple cu site-uri web pe care le recomand pentru documentație C++ Builder]

[1] Programare C++ Builder – Windows Controls <http://www.functionx.com/cppbuilder/>

[2] Dezvoltare aplicație cu interfață grafică - <http://docwiki.embarcadero.com/RADStudio/Tokyo/en/VCL>