Specificatii software

**GGame**

Andrei Marius-Cristian

E 211 A

Contents

[Scop 3](#_Toc478419853)

[Definitii 3](#_Toc478419854)

[Prezentare generală a sistemului 3](#_Toc478419855)

[Perspective 3](#_Toc478419856)

[Cerinte de performanta 4](#_Toc478419857)

[Fiabilitate si constrangeri 4](#_Toc478419858)

# Scop

Scopul acestui document este de a oferi o descriere detaliata a soft-ului GGame. De asemea prezita constrangerile, interfata si interanctiunea cu alte software-uri.

# Definitii

|  |  |
| --- | --- |
| API | Application Programming Interface |
| user friendly | Software usor de folosit |
| Sprite | Reprezentarea vizuala a unui obiect in joc |
| ASCII | Codul Standard American pentru Schimbul de Informații |
| UI | Interfata utilizatorului |

# Prezentare generală a sistemului

GGame reprezinta o unealta pentru dezvoltare jocuri folosind interfata consolei. Scris in limbajul C++, ajuta utilizatorul cu un set de clase si setari user friendly. Defineste clase prestabilite ce ajuta ulizatorul sa dezvolte un joc dupa standardele POO. Foloseste WIN API pentru preluarea Inputului si manipularea consolei.

API-ul vine la pachet cu o aplicatie care genereaza Sprite-uri pe baza unui fisier .bmp si le salveaza in memoria nevolatila pentru a putea fi incarcate de Ggame. Aplicatia se numeste SpriteEditor si nu reprezinta scopul acestui document. Pentru mai multe detalii vizitati [pagina de github](https://github.com/cristyhj/GGame).

Software-ul este destinal platformelor Windows.

# Perspective

Cerinta principala presupune dezvoltarea unui sistem de framework ce updateaze output-ul visual. Clasa principala se va numi GameObject si orice alta clasa care va mosteni clasa GameObject va avea un set de metode disponibile pentru overload. Aceste metode vor avea functionalitati diverse.

Visual, obiectele se vor afisa prin intermediul clasei Sprite. Aceasta va contine un set de charactere ASCII, asezate sub o anumita forma precum si alte detalii despre pozitia pe ecran.

Aplicatiite vor putea integra sunet prin intermediul clasei Audio. Aceasta va manipula fisiere de tip .wav .

Inputul va fi preluat de la tastatura prin intermediul clasei Input. Aceasta foloseste in spate WIN API si constrange utilizatorul la folosirea platformei Windows.

Setarile consolei si manipularea acesteia vor fi prelucrate de clasa Console. Aceasta foloseste de asemenea WIN API.

Fizica ce reda miscarile va fi implementata in clasa Physics. Orice obiect care va contine o instanta a clasei Physics va fi afectat in mod direct la fiecare updatare.

Clasa UI se va ocupa de afisarea informatiilor in timp real si nemijlocit sub format text pe ecran.

# Cerinte de performanta

Performanta jocurilor va depinde de dezvoltarea specifica nu de API-ul GGame.

Performanta in timp:

MUST: 60 FPS in medie

PLAN: 100 FPS in medie

WISH: 120 FPS in medie

Valorile nu trebuie a fi depasite nici in cazurile extreme.

# Fiabilitate si constrangeri

Imaginile ce pot genera Sprite nu pot fi decat in format .bmp .

Sunetele nu pot fi decat in format .wav .

API-ul este destinat platformei Windows.

Orice compilator de C++ poate fi folosit.