

Report
“MTSK-GAME”

Pietro Olivi
Leonardo Tassinari
Lorenzo Dalmonte
Alessio Paoloni

December 24, 2022

Contents

1	Analysis	2
1.1	Requirements	2
1.2	Domain analysis	3
2	Design	4

Chapter 1

Analysis

The software commissioned by the teachers of the Object Oriented Programming course aims to create an application designed to enhance psychomotor skills through a gaming experience based on multitasking.

The term *Multitasking* refers to the ability of a person or a product to do more than one thing at a time.

1.1 Requirements

Functional

- Upon starting, the software will display a simple minigame.
A minigame is a never-ending challenge that requires simple actions from the player in order to keep the game going on. After a short amount of time a new minigame will show up and so on until all four minigames are displayed. The player's goal is to last as long as possible. After failing a minigame the application will display the final score.
- The software has to keep track of how long the player lasts in the current match in order to calculate the score.

Non functional

- The application shall sustain high framerate (around 120 fps) in all sections of the gameplay, even on older hardware¹.
- It shall be possible to easily develop and swap the minigames among the ones that best fit the training purposes of the user on top of those

¹e.g. Intel Core i3 (fourth generation), 4Gb of RAM.

already provided.

1.2 Domain analysis

MTSK-Game shall display some *minigames*, The provided minigames are:

- *WhacAMole* where the player has to stomp appearing *moles* before they get away.
- *CatchTheSquare* where the player must collect *squares* running over them with a *circle* before they disappear.
- *DodgeATriangle* where the player has to slide a *rectangle* up and down, switching lanes to avoid hitting moving *triangles*.
- *FlappyBirdAlike* where the player needs to control a *cursor* leading it to fit between *obstacles* that will come towards it.

Upon user defeat, the *score* will be shown.

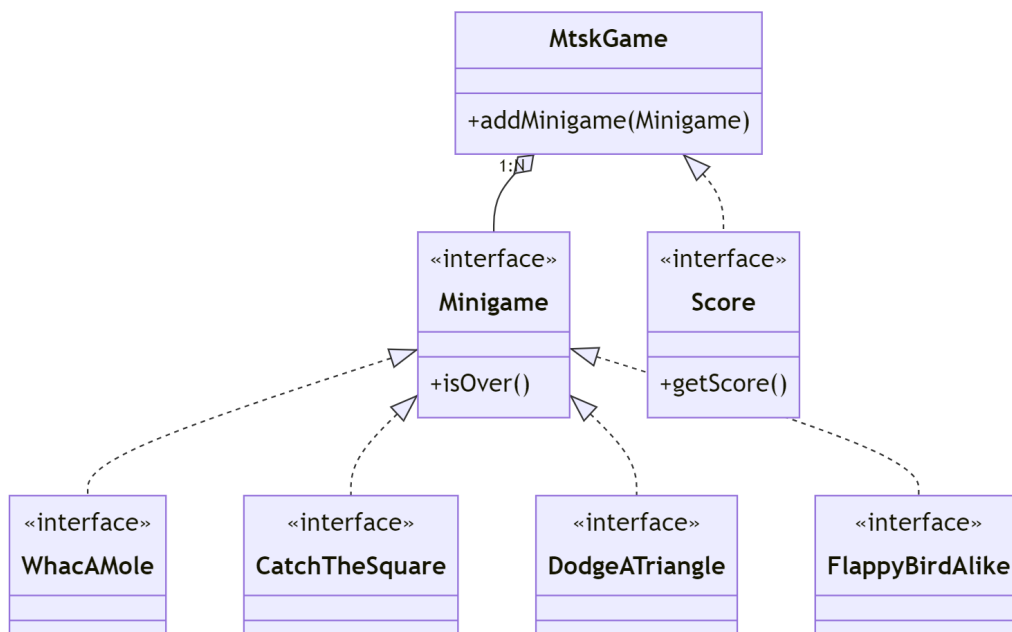


Figure 1.1: UML diagram of the domain analysis

Chapter 2

Design