

Design Overview for TetrisSharp

Name: Duong Quang Thanh

Student ID: 104167828

Design Pattern

The Singleton Pattern ensures that only one instance of a class exists throughout the application, such as in the GameState and GameUI classes where only one instance is enforced within the Main method.

The Abstract class Block serves as a foundational class for various Block types such as IBlock, LBlock,.... It encompasses methods for rotation, movement, and resetting, along with properties for tile positions, starting offsets, and unique IDs.

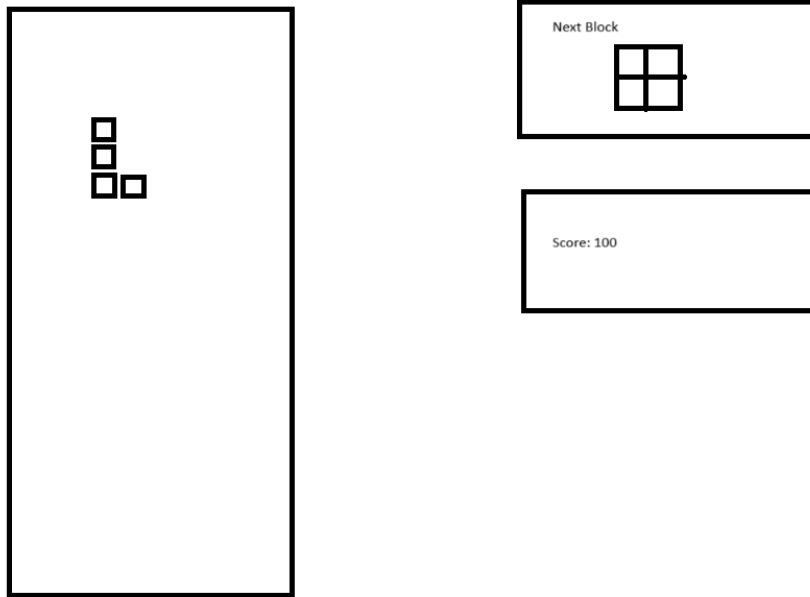
Another design pattern, the State Pattern, is employed to alter an object's behavior based on internal state changes. In this code, the GameState class primarily manages the game's behavior, making it adaptable to different states.

Summary of Program

TetrisSharp is a captivating game inspired by the classic puzzle sensation Tetris, implemented using the SplashKit SDK for graphical user interface and game development. At its core, TetrisSharp features a grid-based game board where the block—comprised of four square blocks—cascade downwards.

Players can easily control these shapes, using keyboard controls for movement and rotation, to form complete horizontal lines. Upon completing a line, it disappears, earning the player points. The game mechanics encompass essential elements such as line clearing, shape manipulation, collision detection, and scoring calculations. With the ability to start and restart the game, players can continuously refine their Tetris prowess and strive for higher scores. TetrisSharp faithfully recreates the beloved Tetris experience, offering both enjoyment and challenge in equal measure.

Here is the sketch of the game:



Required Roles

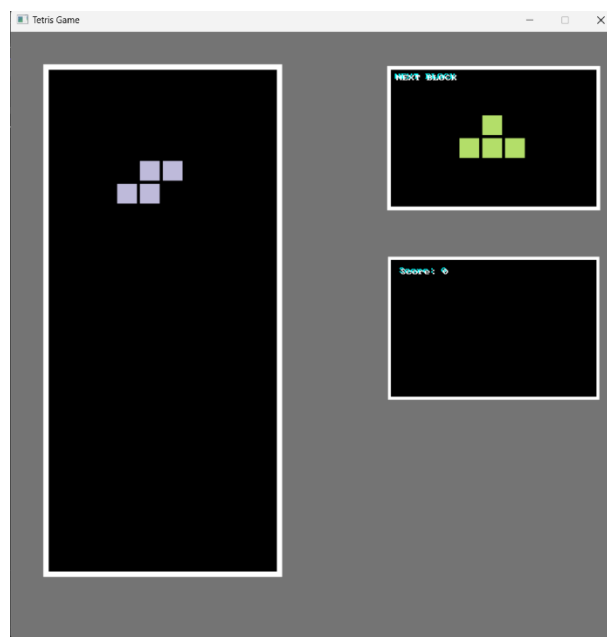
Responsibility	Type Details	Notes
GameGridClass	This class contains column, rows number, color of cell and the methods to check whether that rows is full of blocks or not, it returns the gameboard attributes.	
PositionClass	This class return row and column position.	
BlockClass	This class contains position of the block, tile of block.	
GameStateClass	This class contain the game logic and game implement such as score, clear the completed line, rotate block, move block.	
BlockQueueClass	Due to the block need to change after it placed, so block queue handle the logic of it.	Update HD
UIClass	This class define the position of user interface, which is created from SplashKit SDK. This class return the graphical user interface based on GameGrid created before.	Update HD

Added functionality

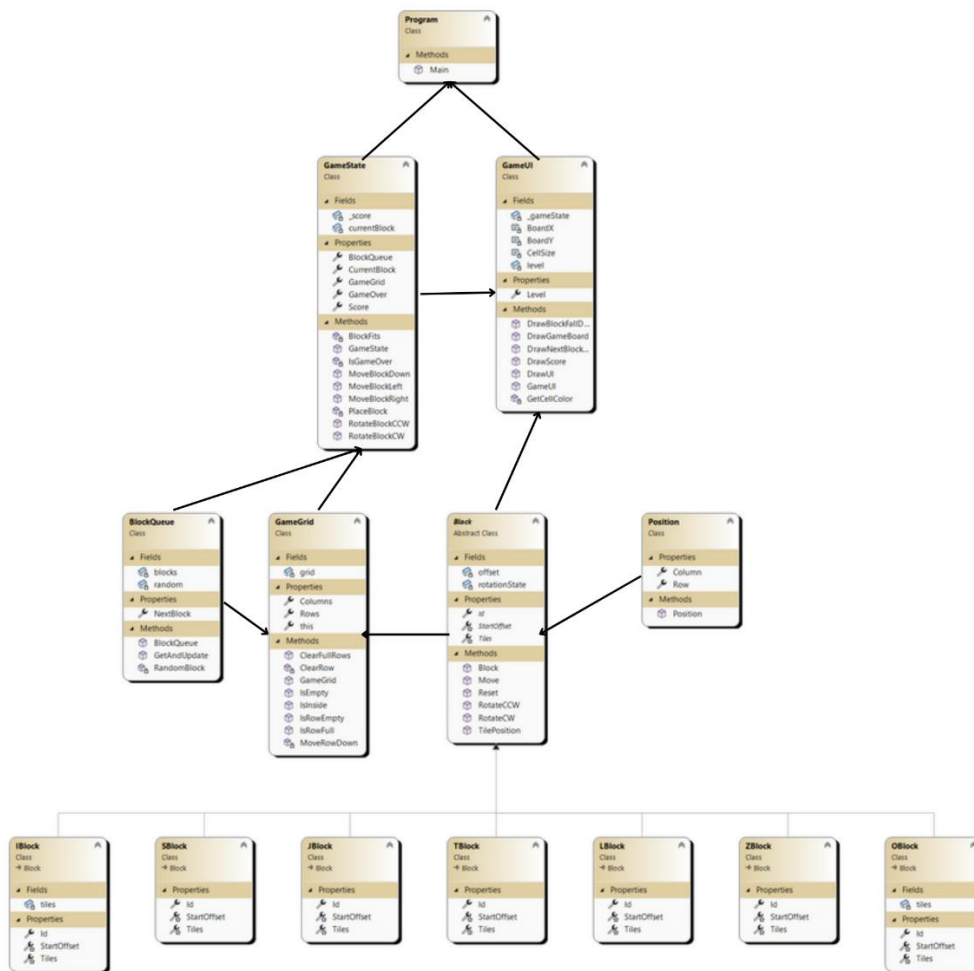
The last version is lack of UI and managing the next block, so in this version, I added the user interface which based on the sketch, and testing the function that remove the line.

Moreover, restart when game over is necessary, so I add function to remove all the placed block and reset the score.

Random Block Generator: The Block queue randomization method involves assembling a collection comprising all seven block shapes, mixing them randomly, and then distributing each shape to the player in sequence to guarantee an equitable distribution of shapes. Once a shape is used, it is slowly drown to the bottom of the bag, preventing its immediate reuse and ensuring a varied gameplay experience without repetitive sequences. This approach ensures that each shape is only utilized once in each cycle through the queue.



Class Diagram



Sequence Diagram

