



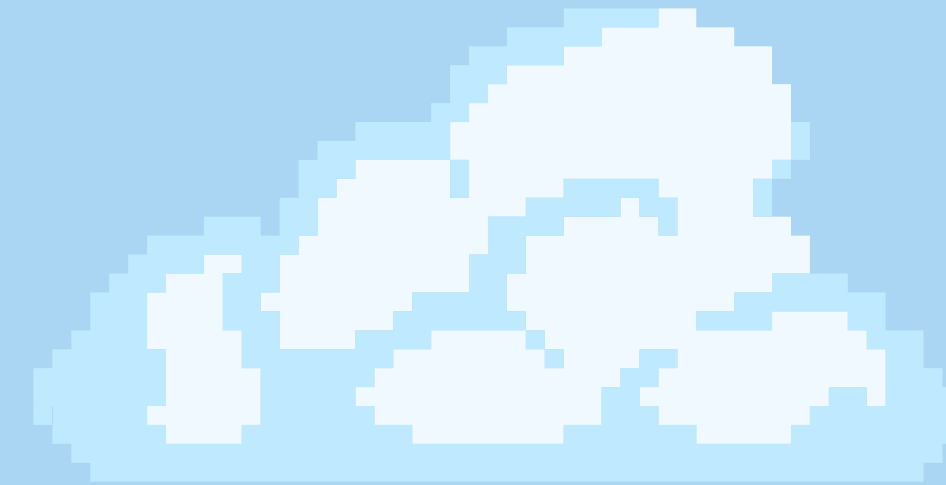
RECYCLED TETRIS

Progetto Vue.js 2 di Anthony Dei Medici

TRY ME LIVE

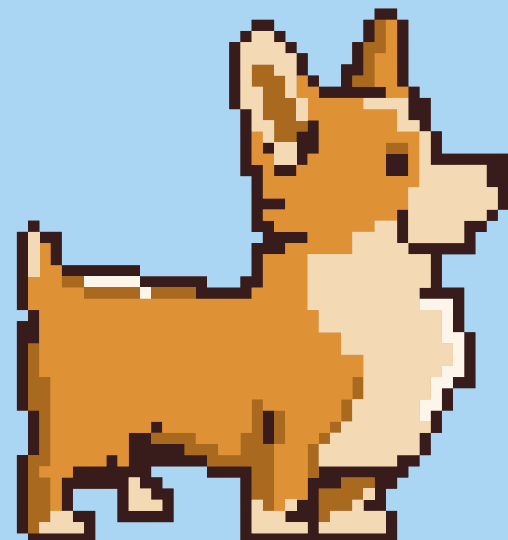
Github repo: <https://github.com/AnthonyDM-Dev/Recycled-Tetris>

SUMMARY



<u>Mission</u>	<u>Pag. 03</u>
<u>App features description</u>	<u>Pag. 04</u>
• <u>Start menu</u>	<u>Pag. 05</u>
• <u>Game page</u>	<u>Pag. 06</u>
• <u>Pause menu</u>	<u>Pag. 07</u>
• <u>Controls</u>	<u>Pag. 08</u>
• <u>Facts popup</u>	<u>Pag. 09</u>
• <u>Level up</u>	<u>Pag. 10</u>

<u>Folders structure</u>	<u>Pag. 11</u>
• <u>Components</u>	<u>Pag. 12</u>
• <u>Tetris folder</u>	<u>Pag. 13</u>
• <u>Styles folder</u>	<u>Pag. 14</u>
• <u>Static folder</u>	<u>Pag. 15</u>
<u>Events handling</u>	<u>Pag. 16</u>
<u>Difficult aspects</u>	<u>Pag. 17</u>
<u>Contacts</u>	<u>Pag. 20</u>
<u>Thank you</u>	<u>Pag. 21</u>




MISSION

REALIZE A MINI GAME TALKING ABOUT SUSTAINABILITY

The focal point is to raise awareness
about plastic pollution by playing Tetris.



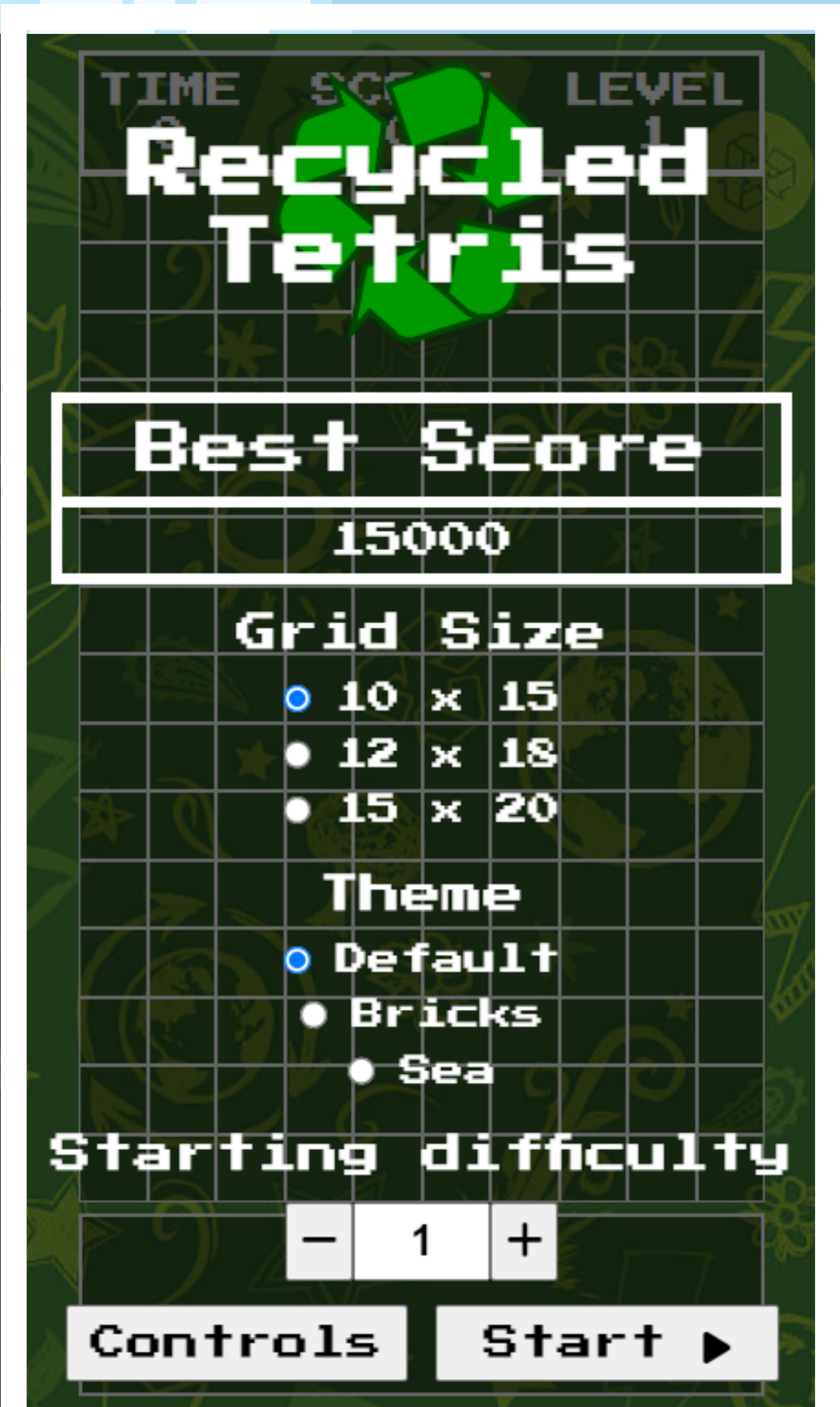


APP DESCRIPTION

START MENU

The first page of the app is a popup that disappear once the player has set all the game configuration and pressed "Start".

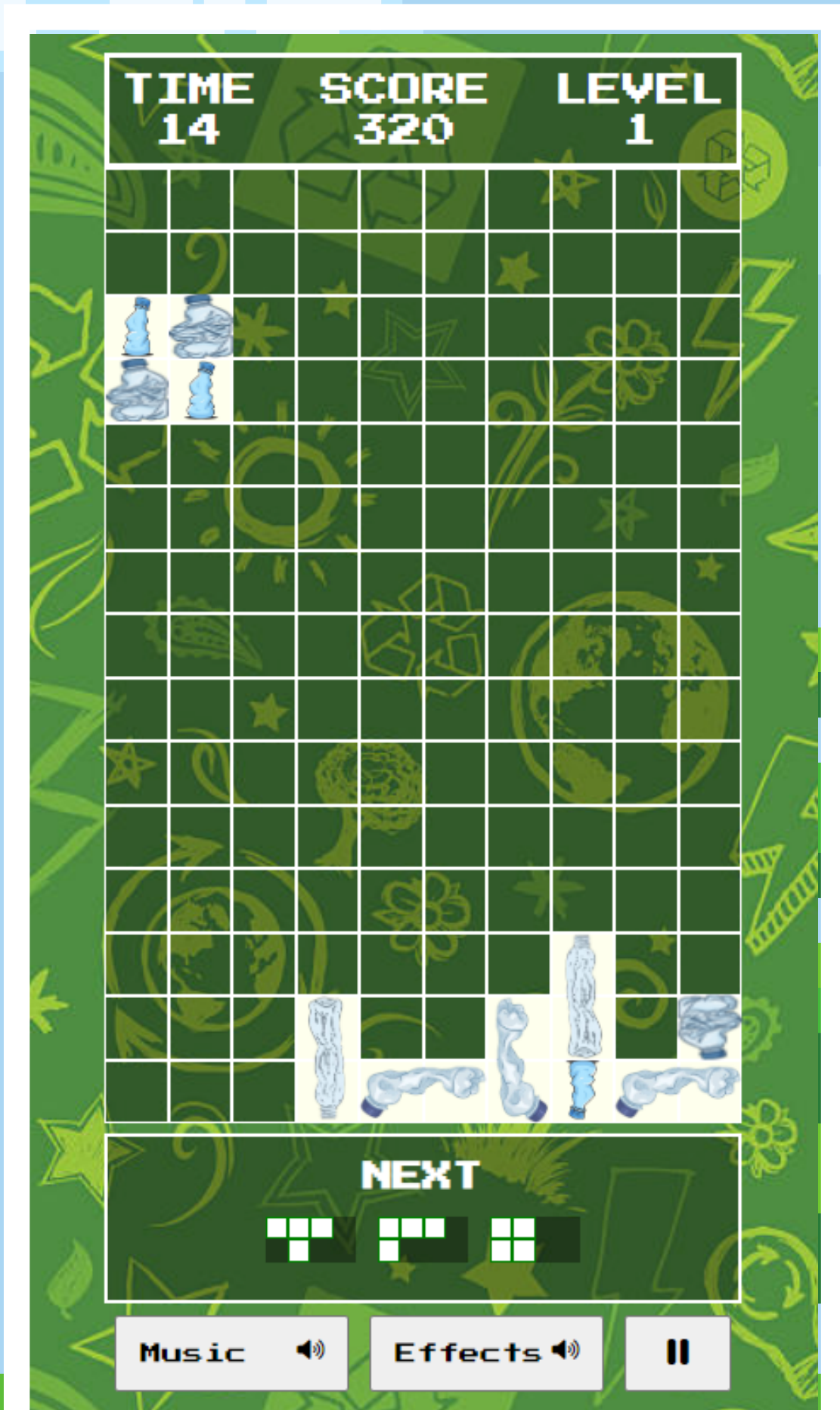
Also, a "Controls" button appear to give to the player the instructions about keys and gestures.



GAME PAGE

This is the main App page.

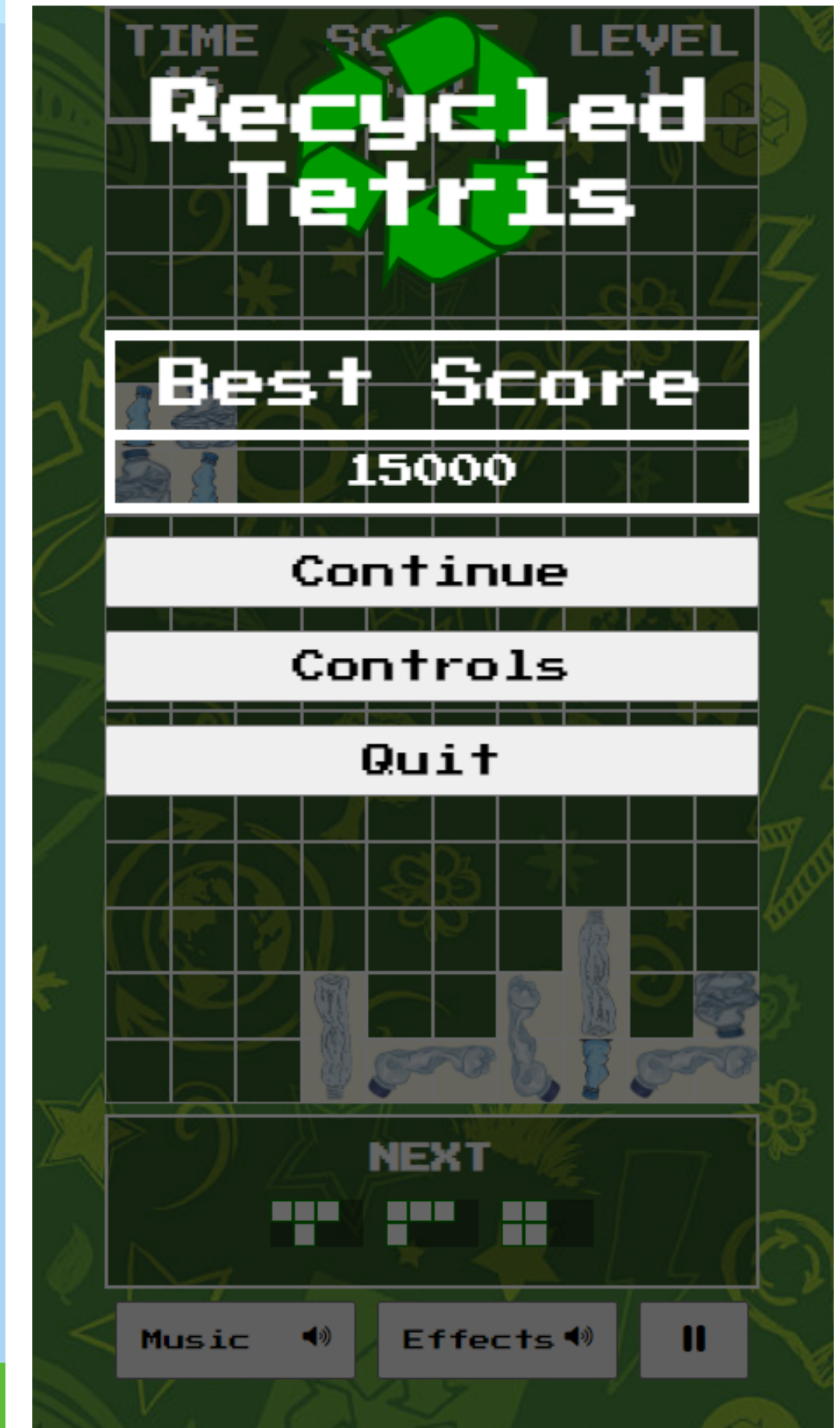
Here, the user can play tetris, have a preview of the incoming pieces, have a reminder about the score, time and difficulty level, activate and deactivate music and sound effects and pause the game.



PAUSE MENU

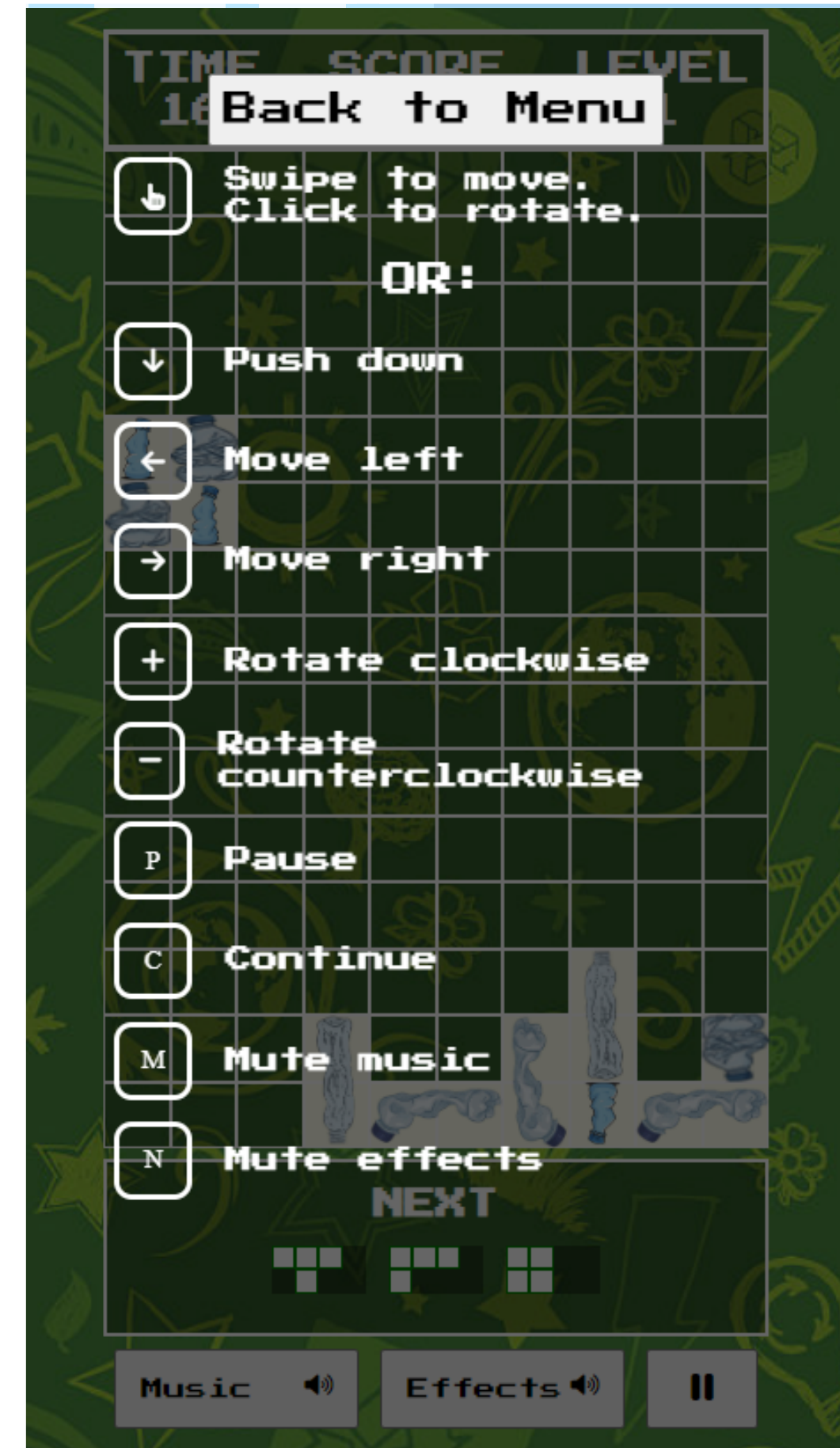
Pausing the game, the player can have a review about the last best score, continue or quit the match and review the game controls.

In this mode, all the blocks cannot be moved or rotated.



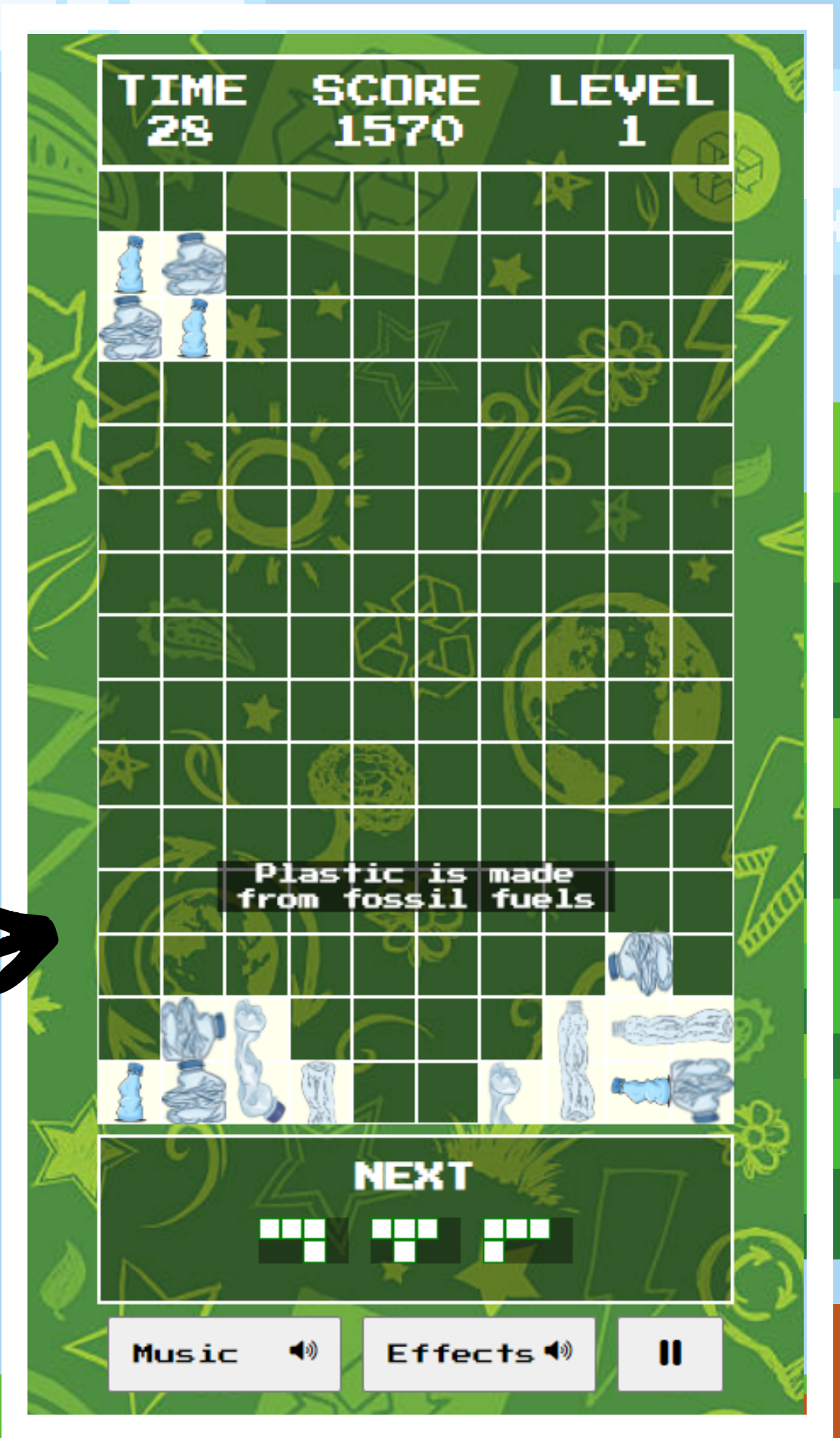
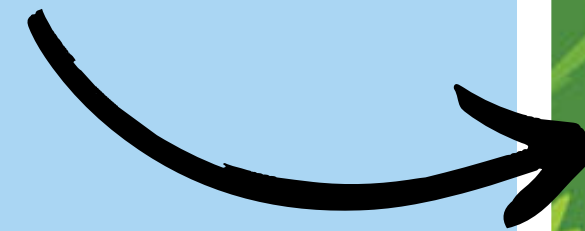
CONTROLS

By clicking on "Controls" button, a small component is called to give the player a small review about keyboard buttons and gestures that can be used in the game.



FACTS POPUP

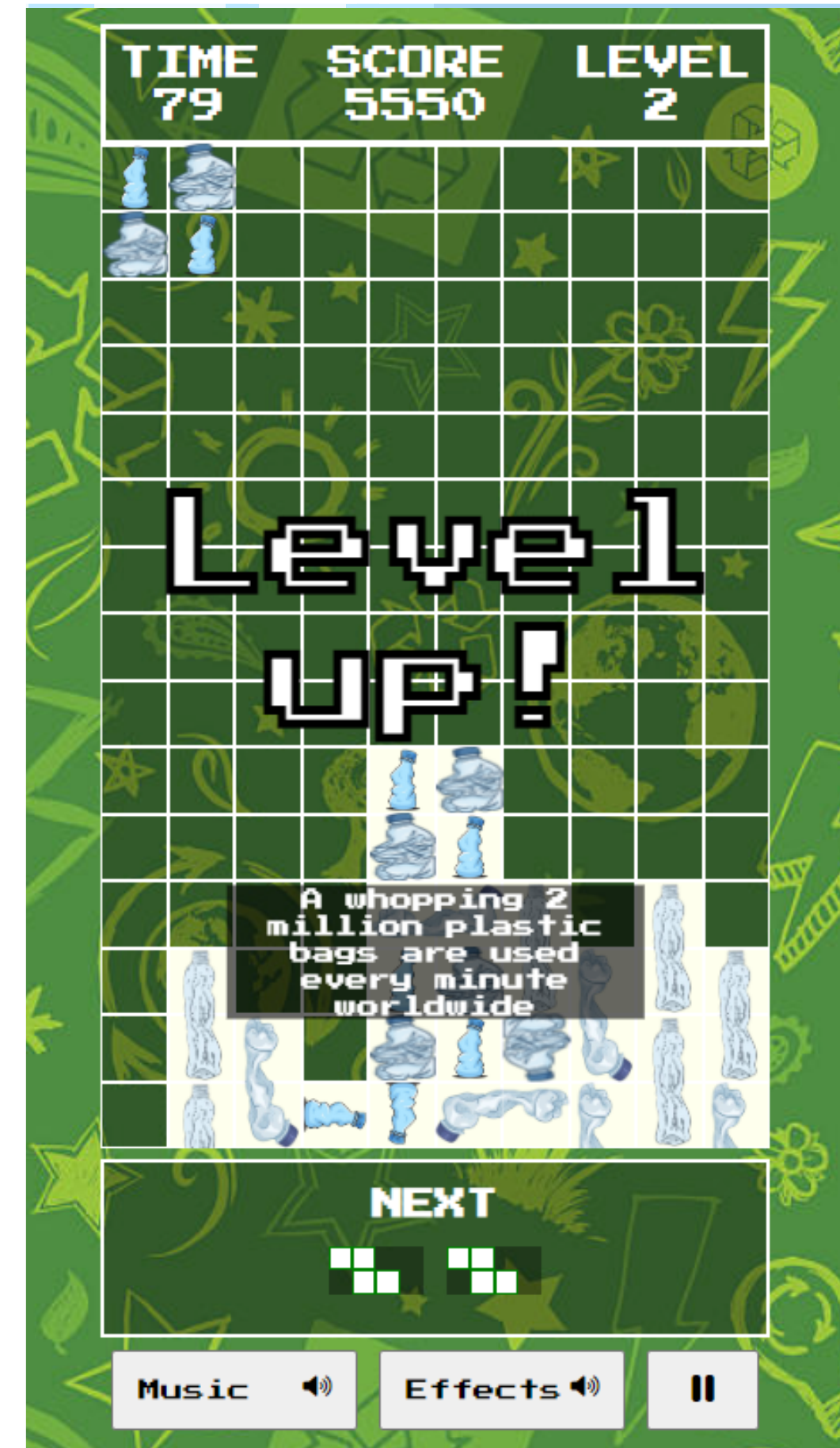
Each time the player delete a row, a small popup appear in the center-bottom part of the screen containing useful information about plastic pollution



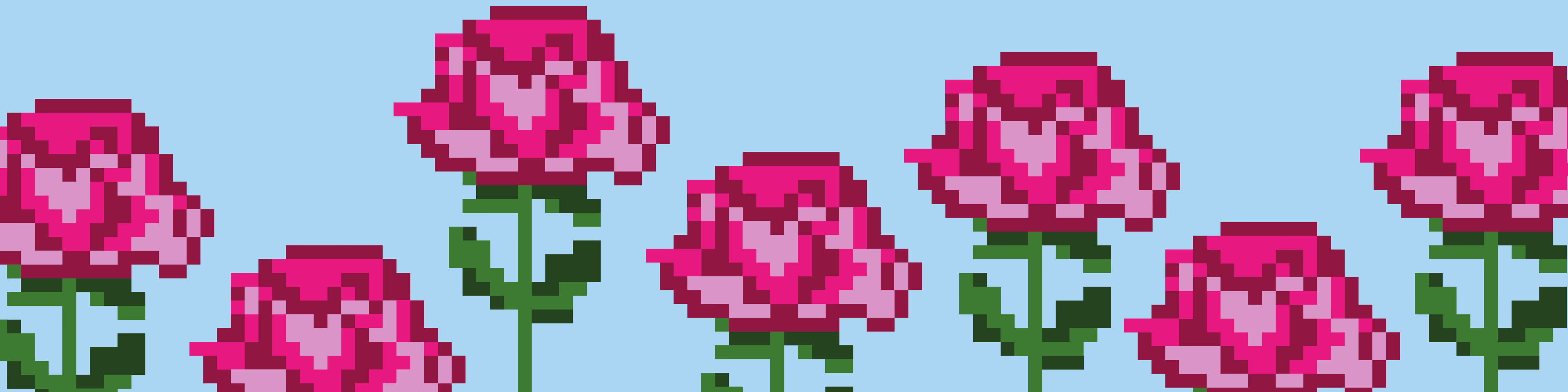
LEVEL UP

Also, each time the player levels up a big reminder appear in the middle of the page.

The player levels up every time he scores 5000 points.



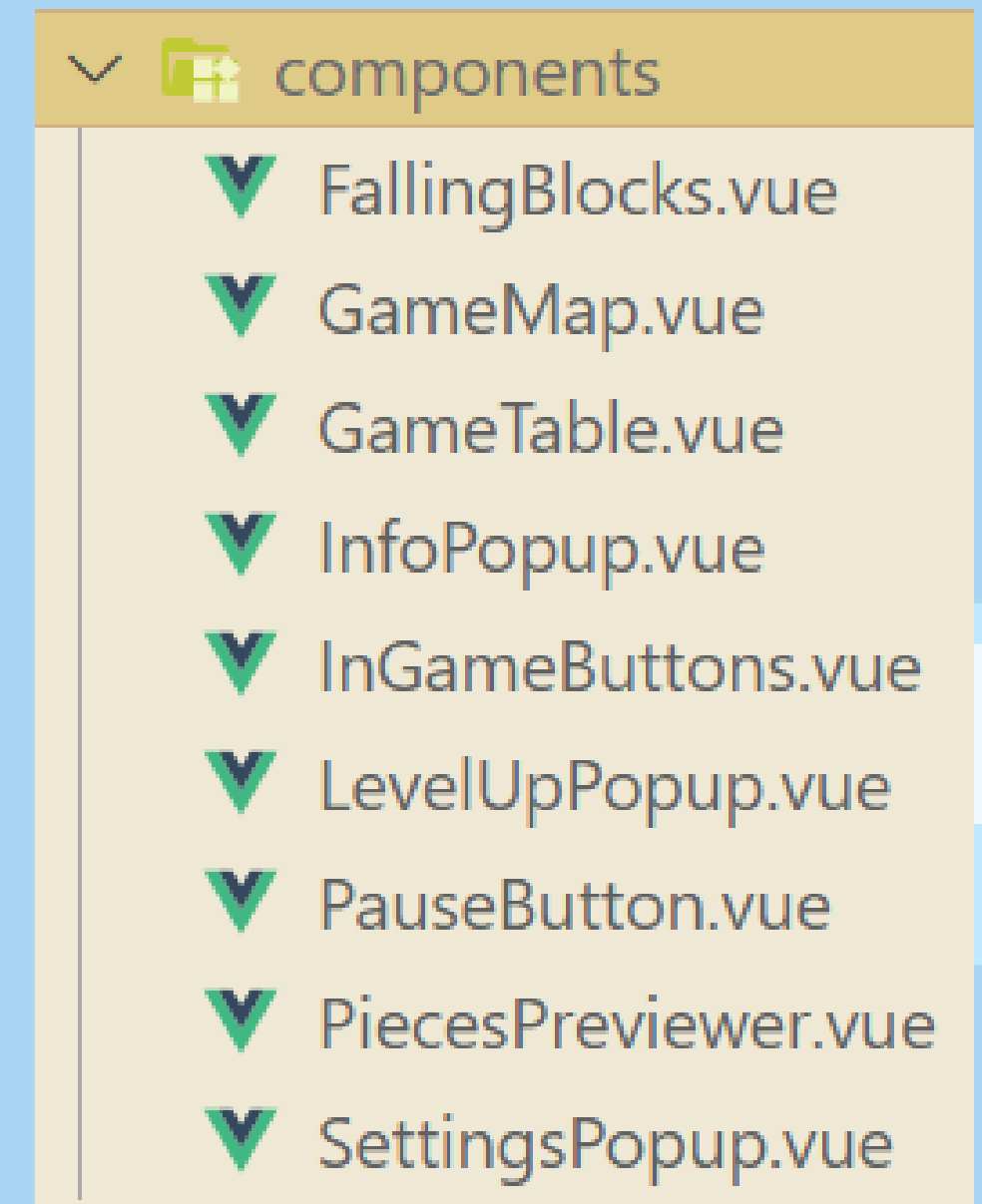
FOLDERS STRUCTURE



COMPONENTS

Components folder lets you increase readability of your app by splitting the full code in multiple components depending on their functionalities:

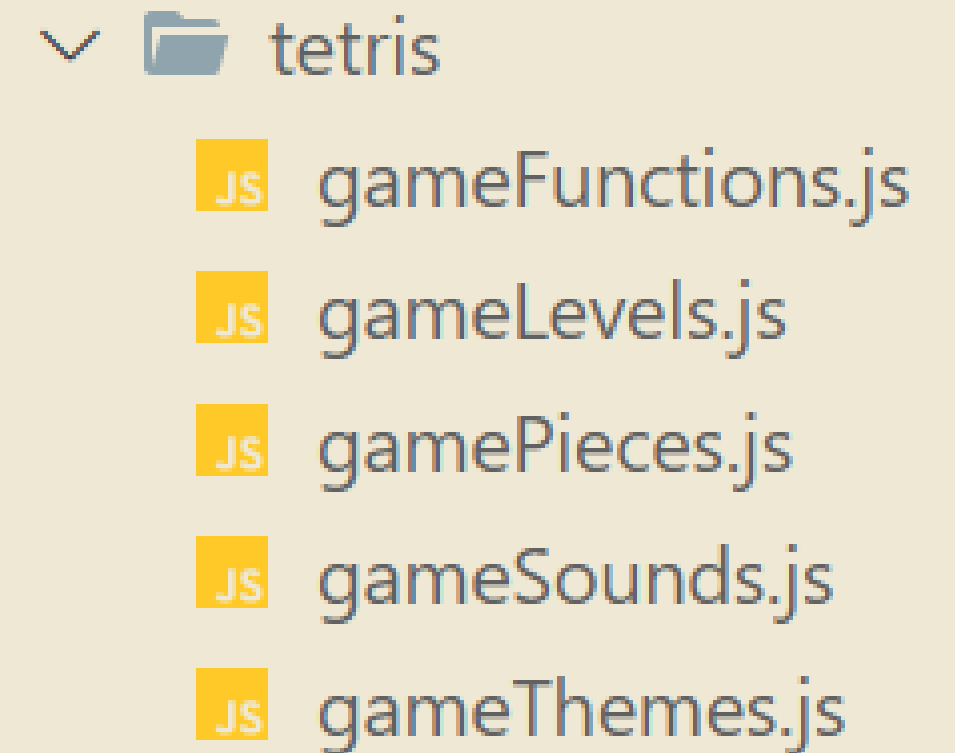
- **InGameButtons:** Game buttons while playing.
- **FallingBlocks:** Containing info about all Tetris blocks in game.
- **GameMap:** Map of the game. Store all the cell coords.
- **GameTable:** Section containing useful game info like Time, Score and Difficulty Level.
- **InfoPopup:** List of facts about pollution appearing every time a row of blocks has been deleted
- **LevelUpPopup:** Popup appearing every time player levels up.
- **PauseButton:** Pause button.
- **PiecesPreviewer:** Preview section of the incoming pieces
- **SettingsPopup:** Popup appearing before start and in-pause game.



TETRAIS FOLDER

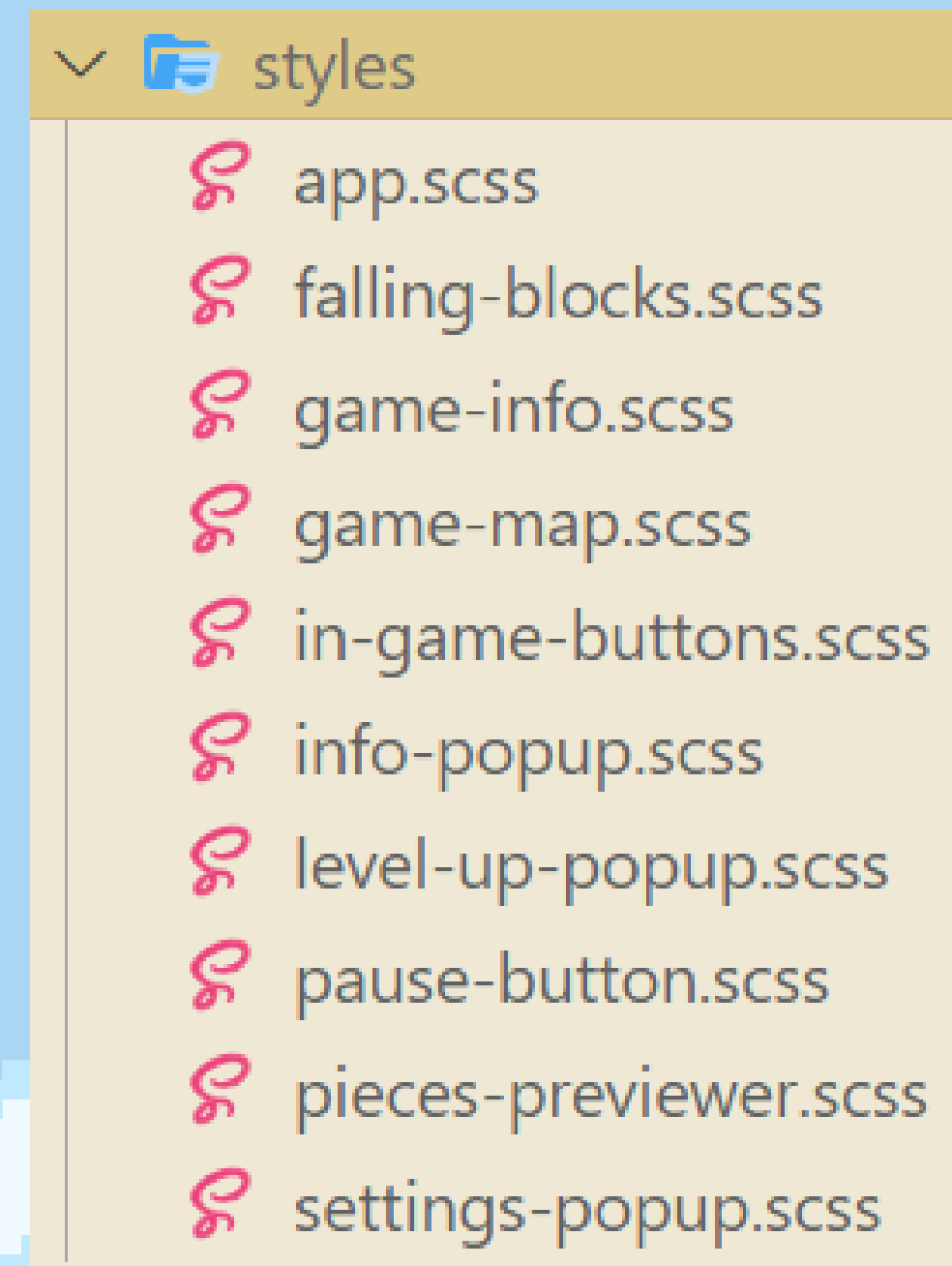
Another way to increase readability and modularity is to separate all game functions in other js files.

- **gameFunctions:** All the functions used to process the game.
- **gameLevels:** Data file containing all useful information about each level.
- **gamePieces:** List of all the pieces appearing randomly during the game.
- **gameSounds:** All the sounds used associate to every player action
- **gameThemes:** Switchable themes offered before starting the game.



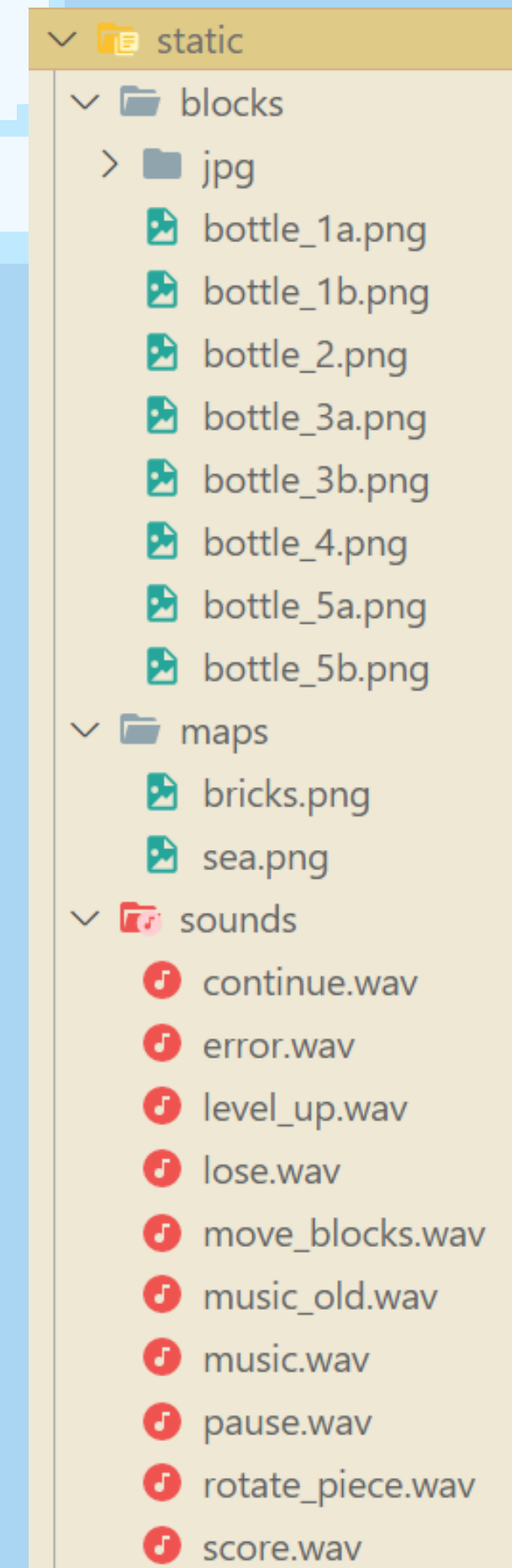
STYLES FOLDER

In order to reduce the lines of code in every .vue file, also is useful to separate all the styling files in a separate folder. Always for readability reasons, file naming is the same as the component who is related to.



STATIC FOLDER

Modularity turns useful also for static files. All the media (images & audio files) used in the game has been stored in different files depending on their functions



EVENTS HANDLING

All of the events are stored in `mounted()` section of the `App.vue` file.

I choose `mounted()` in order to have access to DOM functions and store all of the events in one single place. I divided the events between keyboard events (desktop devices) and pointer events (mobile devices).

- **keydown event:** Used for moving and rotate the blocks with the arrow buttons.
- **keypress event:** Use to give more options for rotating blocks, trigger music, effects and pause/continue the game using keyboard.
- **pointerdown event:** Used to calculate the coords before the finger swipe.
- **pointerup event:** Get the final coords of the swipe and calculate the block movement as a consequence.

DIFFICULT ASPECTS

PART 1

One of the most difficult aspects I had to deal with was the ability to create a function that can rotate every block of every shape around a center point.

And this center point should also be dynamic.

For example, a shape long not more than 3 squares should be rotate in a 3x3 area and a longer shape should be rotate in a wider area (see the examples below)

This made me realize that I'm much better in logical thinking than geometry...

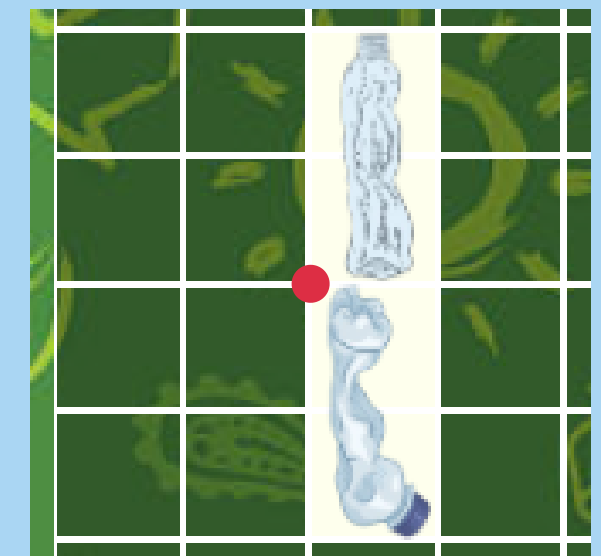
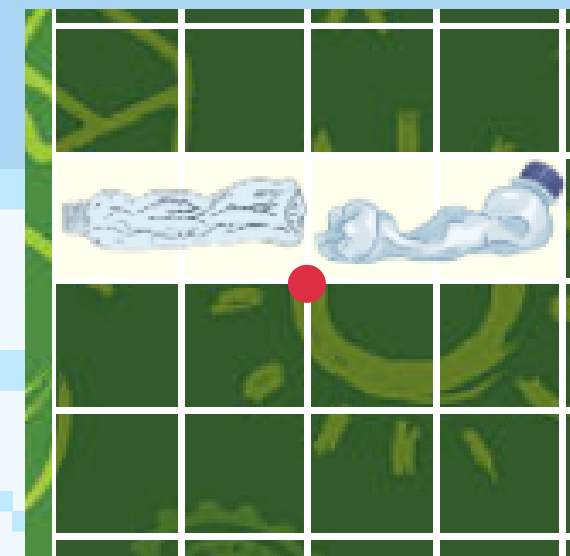
Rotation in 3x3 area



Rotation in 3x3 area



Rotation in 4x4 area



● = center point of rotation

DIFFICULT ASPECTS

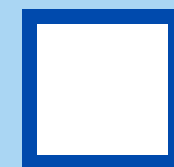
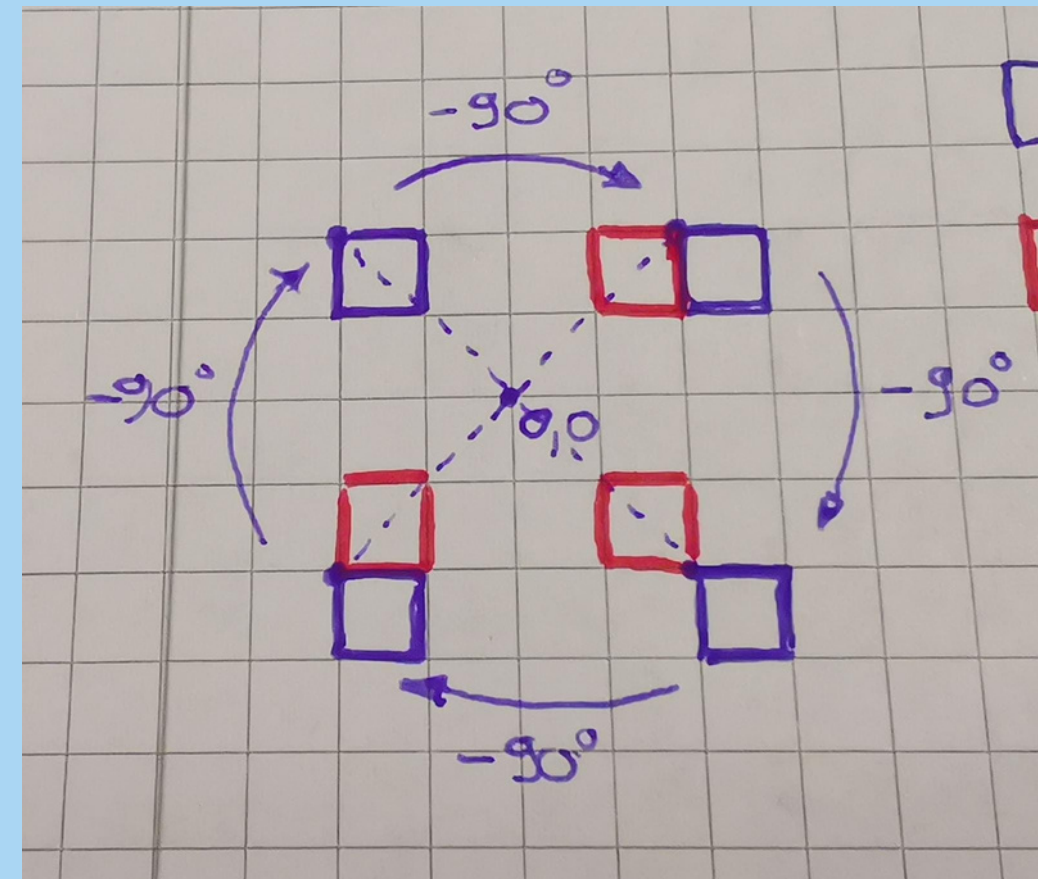
PART 2

The fact is that in my code, every tetris square inside the game map get his position by using "top" & "left" properties.

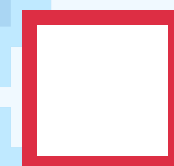
Every time a square rotate around a center point, his "top" coordinate is no "top" anymore.

It was hard to break down in piece what was going on in these steps.

Full rotation of a block



= block position after each rotation



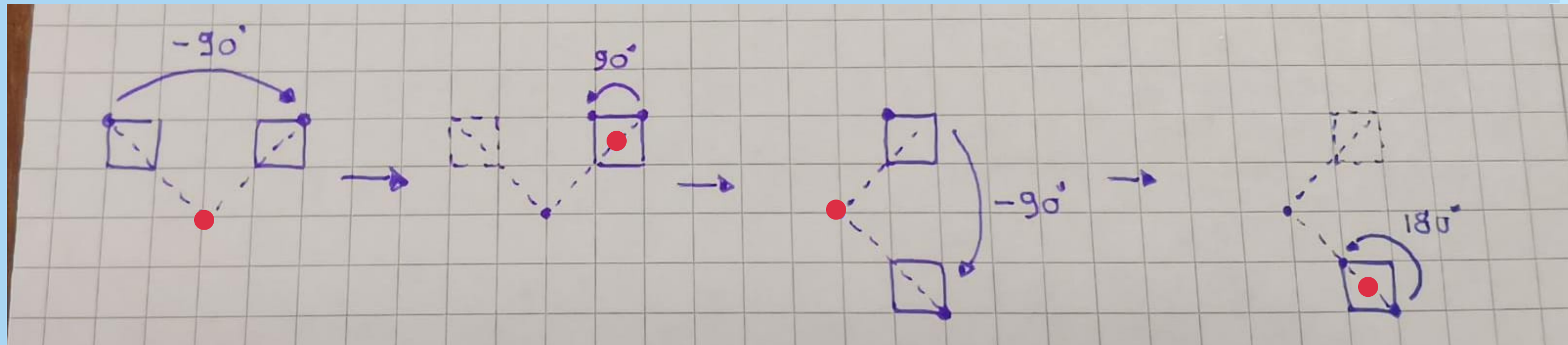
= position where the block should be

DIFFICULT ASPECTS

PART 3

Armed with lots of patience I sketched lots of examples and understood that I have to rotate each square using two different center points and not one in order to get a correct "Tetris rotation".

I tried lots of lines of code before reaching a stable working solution.



Part of the steps required to get a full rotation of a block

CONTACTS

CLICK ON THE ICONS BELOW
AND LET'S KEEP IN TOUCH!



THANK YOU FOR
LISTENING!

