

Начинот на играње

1. Цел на играта

- Играчот треба да го обои целото поле (табла со 5x5 квадрати) користејќи 4 различни бои.
- Бојата на еден квадрат се менува со клик на него откако ќе се избере боја од редот на дното.
- Во режимите „Medium“ и „Hard“, не е дозволено соседни квадрати (овде спаѓаат и дијагоналните соседи) да имаат иста боја.
- Во режим „Easy“, ограничувањето за дијагоналните соседи не важи (само соседи лево, десно, горе, доле).

2. Како се започнува играта

- Кога играта ќе започне, ќе бидат прикажани три копчиња: **Easy**, **Medium**, и **Hard**. Играчот избира ниво на тежина.
- Потоа се започнува тајмер, кој варира во зависност од тежината:
 - **Easy**: 90 секунди
 - **Medium**: 60 секунди
 - **Hard**: 30 секунди

3. Како се завршува играта

- Ако играчот успее да ги обои сите квадрати без да ги прекрши правилата (и без да истече преостанатото време), тој победува.
- Ако истече времето пред да се ги обои коректни сите квадратчиња, играчот губи.

4. Како да се продолжи со играње

- По победа или пораз, играчот има можност да започне нова игра со клик на копчето **Play Again**.

Имплементацијата на функционалностите

1. Генерирање на таблата

Таблата е претставена како матрица:

```
board = [[WHITE for _ in range(COLS)] for _ in range(ROWS)]
```

Секој квадрат на таблата има почетна боја (WHITE), а може да се ажурира со избрана боја од последниот ред бои.

2. Избор на боја

- На дното од екранот е прикажана редица со четири бои. Играчот може да кликне на бојата за да ја избере.
- Избраната боја се означува со бела рамка околу неа.

```
color_index = x // SQUARE_SIZE
if color_index < len(COLORS):
    selected_color = COLORS[color_index]
```

•

3. Ограничувања за соседите

- Во функцијата `is_valid`, се проверува дали соседите (хоризонтални, вертикални, или дијагонални) имаат иста боја:

```
neighbors = [(-1, 0), (1, 0), (0, -1), (0, 1)]
if difficulty in ["Medium", "Hard"]:
    neighbors += [(-1, -1), (-1, 1), (1, -1), (1, 1)]
```

- Оваа проверка е активна само во режимите „Medium“ и „Hard“.

4. Тајмер

- Кога започнува играта, се зачувува времето на почеток:

```
start_time = time.time()
```

- Во текот на играта, се пресметува преостанатото време:

```
elapsed_time = time.time() - start_time
remaining_time = max(0, time_limit - elapsed_time)
```

5. Победа или пораз

- Победата се проверува преку функцијата `is_game_won`, која верифицира дека:
 - Сите квадрати се обоени (не се бели).
 - Сите правила за соседи се исполнети.
- Ако истече времето, `game_lost` се поставува на `True`.

6. Повторна игра

- Кога играчот ќе кликне на копчето **Play Again**, сите параметри на играта се ресетираат:

```
board = [[WHITE for _ in range(COLS)] for _ in range(ROWS)]
game_started = False
game_won = False
game_lost = False
```