Wordle Algorithm

General Rule

Wordle is a game which the player has to guess the secret word in **6 tries or less**. If the player guesses a word, which is not recognized or not available in the dictionary, he will be prompted to enter again.

If he guesses the wrong word, different colors will be shown.

If the letter is **the correct letter in the correct position**, color '**Green**' will be shown. For example ...

- If the correct word is '**Planet**' and the guess is '**Pastel**', the letters '**P**' and '**e**' background will turn **green**.

If the letter is **the correct letter in the wrong position**, color '**Orange**' will be shown. For example ...

- If the correct word is '**Planet**' and the guess is '**Pastel**', the letters '**a**', '**t**' and '**l**' background will turn **orange**.

If the letter is **wrong**, color '**Grey**' will be shown. For example ...

- If the correct word is '**Planet**' and the guess is '**Pastel**, the letters '**s**' background will turn **grey**.

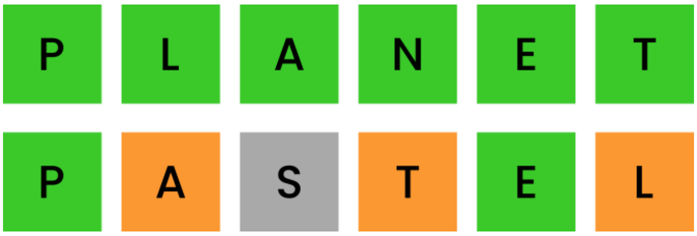

Fig.1 General rule

Duplicated Values

Duplicated values will be checked only once and the rest will be grey out regardless of the position. For example ...

- If the correct word is '**silent**' and the guess is '**sister**', **the first** '**s**' background will turn **green** and **the second** '**s**' background will turn **grey**.

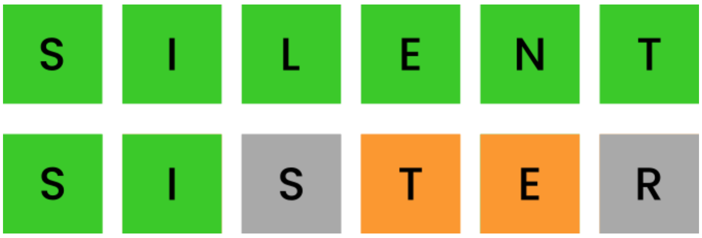

Fig.2 Duplicated letters

- If the correct letter is '**silent**' and the guess word is '**little**', **the first 'l'** and **'t'** background will turn **orange** and **the second 'l'** and **'t'** background will turn grey.

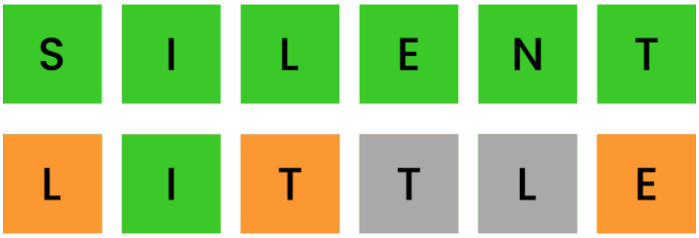

Fig.3 Duplicated letters

Key colors

In addition to the letter background, the colors of the keys pressed are need to be tracked.

- If **the correct letter in the correct position** is typed, the key color will turn **green** and will not change back to any color until the next game.
- If **the correct letter in the wrong position** is typed, the key color will turn **orange** if the key color is not green originally (if the correct letter in the correct position is already guessed previously, the key color will not change back to orange).
- if **the wrong letter** is typed, the key color will turn **grey** only if the key color is not green or orange originally.

  For example …

- If the correct letter is ‘**silent**’ and **the first guess word** is ‘**little**’, **the keys ‘l’, ‘t’** and **‘e’** will turn **orange**. **The key ‘i’** will turn **green**.

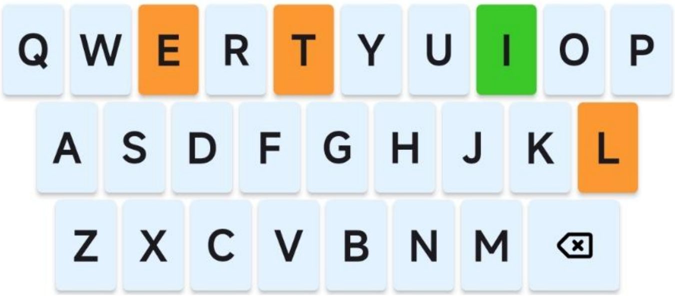

Fig.4 Key colors after 1st guess

- If **the second guess word** is ‘**sister**’, **the keys ‘l’, ‘t’** and **‘e’** will remain **orange** and **the key ‘i’** will remain **green**. **The key ‘s’** will turn **green**, and **the key ‘r’** will turn **grey**.

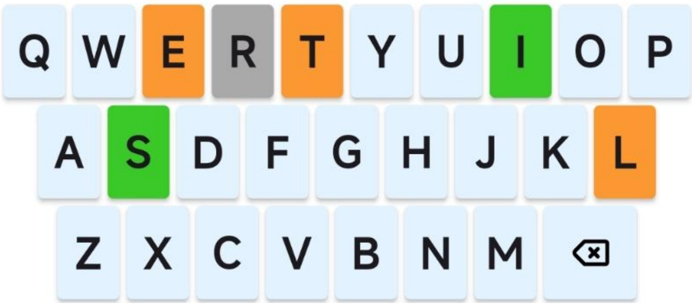

Fig.5 Key colors after 2nd guess

Wordle Algorithm – Htet Wai Hlyan Kyaw

Implementation

To implement this algorithm in Dart, we need to create a two-pass checking algorithm (First loop for the green case and second loop for the orange and grey cases). The algorithm will work like this. Here is simplified answer checking logic for 6 letter words.

```
answer = [’a’, ‘p’, ‘p’, ‘l’, ‘e’, ‘s’]
guess = [’b’, ‘a’, ‘n’, ‘a’, ‘n’, ‘a’]
answer letter count = {‘a’: 1, ‘p’: 2, ‘l’: 1, ‘e’: 1, ‘s’: 1}
// First loop to check matching case
for (current letter in guess) {
    if (current letter in guess == answer letter) {
        change box color to green
        change key color to green
        reduce answer letter count [current letter]
       increase green count
    }
}
//Second loop to check close and wrong cases
for (current letter in guess) {
    if (current box color == green) {skip this iteration}
    if (answer letter count [current letter] > 0) {
        change box color to orange
        if (key color != green) {
            change key color to orange
        }
        reduce answer letter count [current letter]
    }
    else {
        change box color to grey
        if (key color != green and key color != orange) {
            change key color to grey
        }
    }
}
if (green count == 6) {
    win
}
else {
    next guess, lose or quit according to the program
}
```