# Build a Prototype

A Midnite Game Development test

## Introduction

Time is a factor in this test.
We're more likely to receive a working and bug free partial prototype, than a complete one that has tons of issues.

To submit the test, you have to create a public repository (GitHub or Bitbucket). Commit each task at least once and release the final prototype as .apk.

A brief documentation that explains your choices will be highly appreciated.

---

## Tiles piles:

### Task A

It is common for us, to work on prototypes that share a lot of similarities with already successful games. Let's start of making a prototyped replica.
For this exercise you should build a simplified version of **Sandwich** ([Android](), [iOS]()).

The player has to build a Sandwich, flipping the ingredients one by one.
Ingredientes can be flipped only over other ingredients.
The puzzle is completed once all the ingredients are stacked together and the bread pieces are on the top and bottom of the pile.
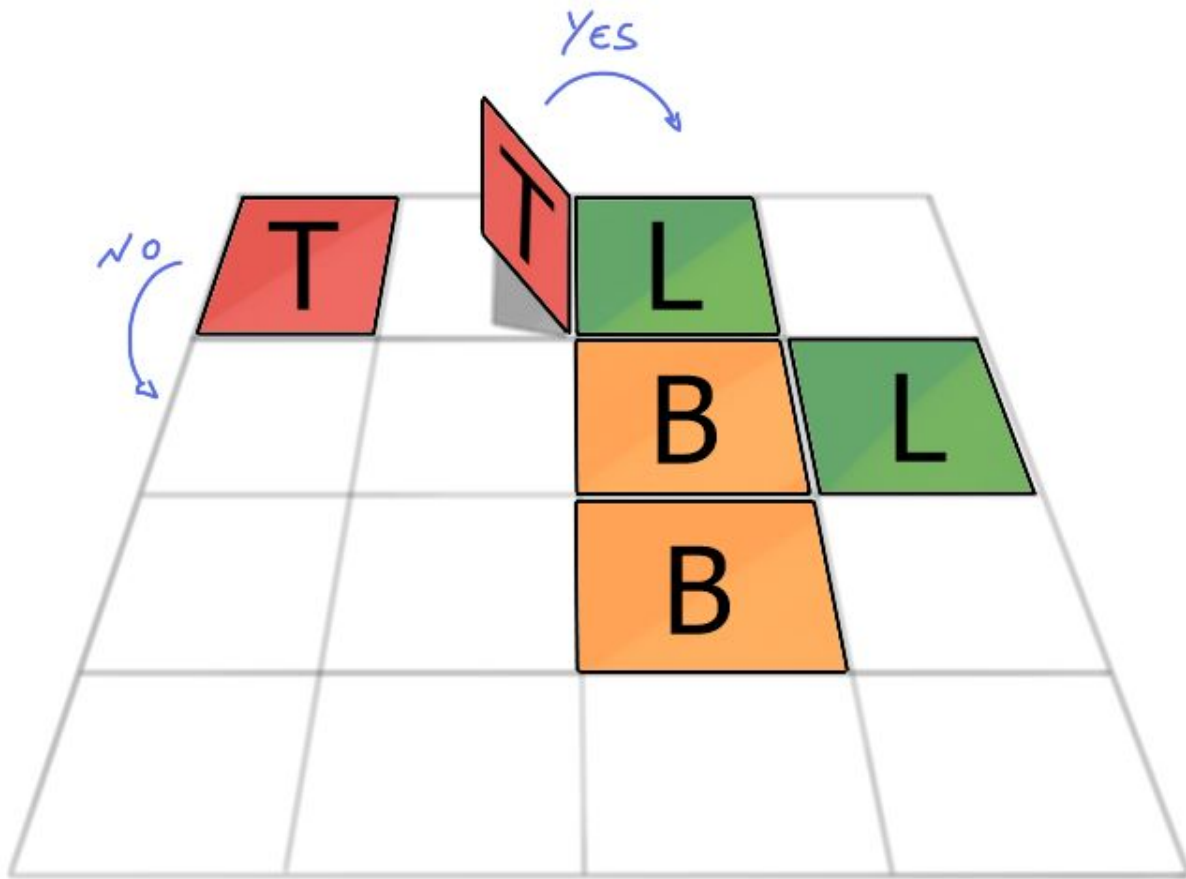
You can build the prototype with given placeholders.

The prototype **MUST FEATURE**:

- A 4X4 Grid.
- 2 Bread pieces.
- At least 2 different ingredient pieces.
- The same control system of the reference. ([iOS](#), [Android](#))
- The same victory condition.
- A reset button (UI) to repeat the game.

What we do **NOT NEED**:

- Metagame mechanics (ingredient shop, city building, etc…).
- Currency of any kind (coins, keys, etc…).
- Final "tap to eat" feedback.
- Skip / Undo.
- Level progression counter.

## Task B (optional)

Now, we want to make some contents

- An algorithm that generates random levels.
- Additional button (UI) to "generate new level".
- Exposed field in editor to customize difficulty (I.E. amount of additional pieces).
- Level layouts can be saved in a serialized file (choose a serialization of your choice).

## Task C (optional)

Using the result of the task A, let's build a variant of that game.

Instead of having "ingredients", the tiles should now work as the very famous 2048.
So for each different color, there is a different value (all powers of 2).
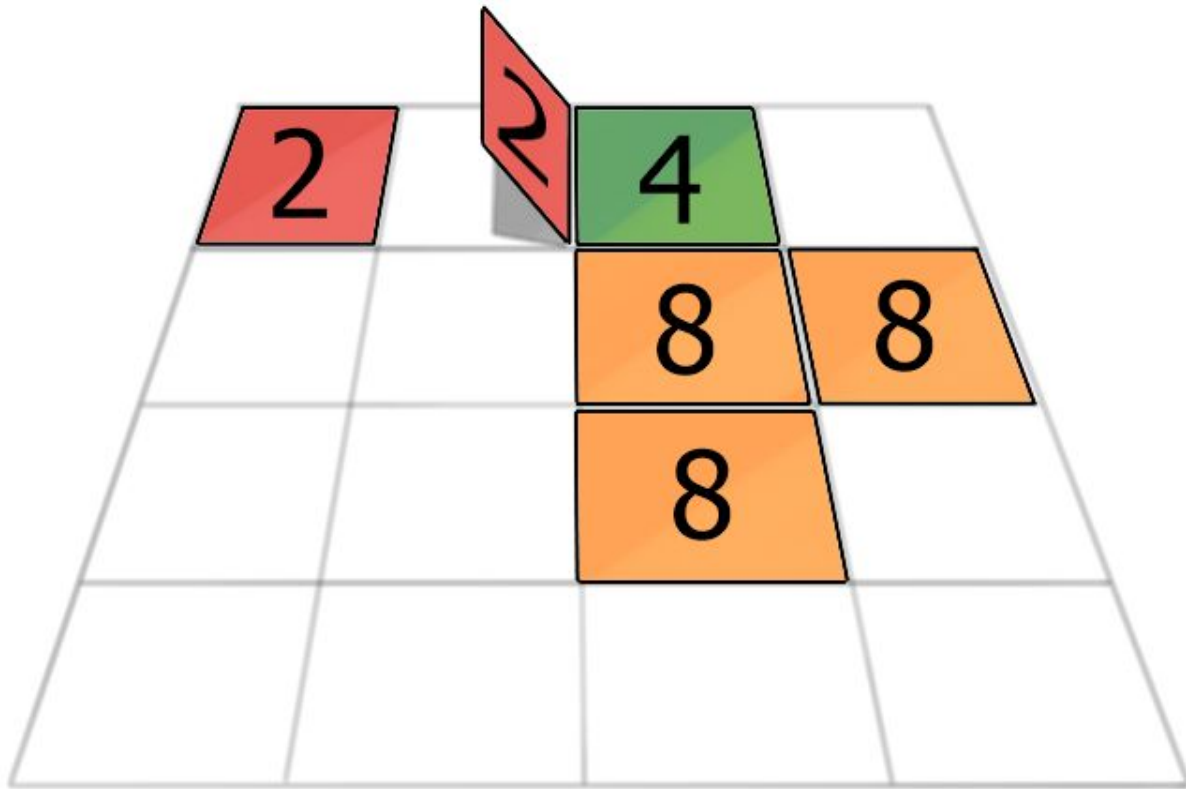
Some of the rules changes:

- When two tiles are stacked one on the other, and they are of the same value, they merge into the next one by value.
  - Note: this could trigger chain reactions in some situations
- The game is won once there is only one tile on the grid.

Perform this task using a FSM or an Event System.

Note:

If you have performed task B, the level generation could work against you. Disable or update it.