

# Requirements for TA Exercise - S2

By group 26 - The Fishermen

For assignment 2, we were asked to split up our `PlayingField` class to reduce its responsibilities. With a sub-exercise of changing our Fish death behavior to a more logical one (eat instead of `setDead`, etcetera).

The requirements are listed in the same way as our game, so we will use the MoSCoW method. First, the functional requirements are listed as must, should, could and won't have. Second, the non-functional requirements are listed.

## Functional Requirements

### 1.1 Must have

- The `PlayingField` must have its responsibilities reduced.
  - To do this, it must be split up into a `GameThread`, a `Renderer` and `PlayingField`.
- The `SinglePlayerPlayingField` must not contain any GUI code. That code must be moved to a GUI class.
- The `EnemyFish` must eat the `PlayerFish` when colliding with it, instead of the `PlayerFish` killing itself.
  - This must also be changed in the sequence diagram.
  - This new behaviour must be tested.

### 1.2 Should have

- The eating behaviour should be implemented using an interface.

## Non-functional requirements

- The naming of the `setDead` methods should be changed to be more logical.
- These new classes will together have at least 75% test coverage.