

In the name of God

## Technical doc

[Here](#) you can see the class diagram of the vision controller.

I tried to make the vision controller codes readable and extensible, for this purpose got help from some principles and design patterns and wrote some comments too, so in this doc, I only mention the main classes and their main behaviours or goals.

# VisionController

It's the main script that controls the vision using other classes and must attach to objects.

At the beginning of the game, in the OnEnable method, it initialises the "Vision" using the "VisionFactory" which is selected in the Inspector.

Then in the Start method, it creates a coroutine and checks the vision and sense area using the "Vision" in every specified second in the Inspector.

# Vision

It is an abstract class of every vision mode that the "VisionController" composites in itself and uses it for drawing the vision and sense area and also handles the logic of the vision.

The "Vision" class provide some methods for it inheritors that handle some common and related logics.

It also has some abstract methods for its inheritors that they must override and implement their own logic because the vision modes have different shapes and calculations.

This abstract class avoid of hard coupling between the "VisionController" and the concrete implementation of vision modes.

# VisionFactory

It is an Implementation of factory pattern to make the vision controller extensible and decouple.

It's also an abstract class that is composited in the "VisionController".