Comments:

Sequence diagram:

The latter is a representation in the form of graphs to explain the operation of our game from the beginning to the end. So from the moment the user starts up until he finishes playing a level. The process repeats this way for each level he wants to play.

Package diagram:

It respectfully represents the various packages used in the code for modeling our game. We have therefore identified 5 main packages, two of which have sub-packages modeled by continuity relations.

Use case diagram:

Used to give a global vision of the different opportunities or actions that the system offers to the user. So the user can start the program then choose a level or click on the Red Cross to quit, usually always available. Then play and at the end either lose or win.

Component diagram:

Describes the organization of the system from the point of view of the modules (view, controller, model, and contract). This diagram makes it possible to highlight the dependencies between the components. Each component provides a very precise service. The view for the graphical interface, the model for the representation way, the controller monitors the respect of the model, the contract to manage the service contracts.

Class diagram:

Subdivided by package, the class diagram represents the different classes used in each of these packages to be able to make instances.