* Work process

To develop this software project, a Waterfall Model was followed as it is explained next.

* 1. Analysis (Knowing the game):

To know the game and have an idea about how to develop the project, two actins were taken:

\* First the physical cards have been checked.

\* Then an app of the game was player.

* 1. Designing the UML:

The UML designing process had 3 versions which are described next:

\*First one with the classes included and the main methods.

\*Second one with less classes to avoid redundancy but more methods.

\*Before designing the last version, it was necessary to try some methods in code to know what would be the practical option, and after that the final version was defined (This is an exception of the engineering software model used).

* 1. Implementation

For coding, the main idea was to follow the design, so, it was started by creating the classes with the main variables and methods and some basic functionality to test the design.

After making some changes in the UML design the main functionality of the implementation was defined and coded, adding some interesting features to make more personal the game (which will be explained later).

Finally, some extra functionalities were added.

In this stage the Waterwall model was combined with a Kanban model in order to apport ideas and work to the team and moreover to work with the testing stage continuously.

* 1. Testing

As was mentioned above, the testing part came together with the implementation work, in order to define what was need to solve, but at the same time it was not a impediment to continue working in another features of the code.

A final test process was followed in separate for each member of the group in order to improve the performance of the code.

* 1. Documentation and Deployment

Finally the documentation stage was set up, creating the required documentation.

* Teamwork

The team work in a collaborative way, starting early the designing and code stages in order to be able to present some extra features and to provide a personalized work.

Every stage has the participation of each member of the team.

* Design choices
* Special elements in your code you want to address and last but not least a small manual for a user to play your game (you can assume this user has played a normal game of Fluxx before)