# 5.1 Planning

The original plan was to develop the whole project in five months in two different stages and documenting the whole process, the first stage was programmed to las two weeks designing and thinking on the objective of the application and making sure that the idea is viable before writing a line of code with a form sent to possible users. The next stage was the development step, first creating the tests that will confirm the application works correctly, this stage was predicted to last 4 months. The following steps can be seen in the Gantt graph on the next page.

After the five months of development, due to external factors, the project was seriously delayed and a redesign had to be done as the complexity of the context manager in the original design made it unviable to deliver for this year. This made it necessary to rethink how to adapt the project to reach the target of September for delivery. In order to reach the objective an extension on the different parts of the project have to be done, this included two more weeks of planning and preparation, twelve more weeks of development where the new configuration and implementation was completed, the rest of the extension was filling out the documentation with the new design.

Even though this was a setback to the project, it allowed gaining experience in how the perception of time in completing tasks has to be refined in order to avoid overextending a project and having inefficiencies.

# 5.2 Budget

In this section, we will explain the budget for the project, which contains the resources used when developing the project. Below is a table with all the details about the cost for the hardware, software and human resources.

[TABLE]

To give some context to the different parts of the table we will explain them further. The reason the cost for the Azure Machine is relative is because it has a pricing system based on pay what you use, meaning that depending on how much you use the resources the pricing varies, in the case of this project the virtual machine was used only when required to minimize costs.

The hardware used didn’t add much cost to the project

* Cheap Hardware
* Why azure is not precise
* More expensive human resources