



END OF STUDY PROJECT

KEEP CALM AND CARRY ON



END OF STUDY PROJECT

Previously, you defined your project and chose your partners. It's now time to start working!



Milestones are defined in Gandalf. Through this agenda, you must work on your project following the specifications defined.

We assume that you will not have enough time to finish!

It's your responsibility to give to the next contributors all the tools, documents and processes that will allow new arrivals to integrate the project.

Speaking of documents and processes ... You'll have to define them during this semester!

Don't lose yourself

In project management, proceeding without any guidelines leaves you in danger of overrunning available resources and failing to achieve your goals. A **project plan** is a specific document which describes how the project will be executed, monitored, controlled and closed.



It may include considerations for:

- ✓ Requirements management (from your specifications defined) ;
- ✓ Scope management ;
- ✓ Schedule management ;
- ✓ Quality management ;
- ✓ Resource management ;
- ✓ Communications management ;
- ✓ Risk management ;
- ✓ Financial management ;
- ✓ Stakeholder management ;
- ✓ Project change management ;
- ✓ ...



As mentionned in the RFC 2119, MAY != MUST.

So, it's up to you to define the needs for your project. Just be aware that we'll expect you to complete most of the elements from the previous list before the end of the ESP!

Deliveries

You are expected to deliver the following files:

- ✓ the source code of your project ;
- ✓ its [updated] specifications (functional and other than functional) ;
- ✓ a document describing your processes and methods (tools, policies, norm, rituals, ...) ;
- ✓ your quality assurance processes and testing ;
- ✓ the user acceptance testing.
- ✓ your hardware needs and specifications



The **InVEST** model should help you assess the quality of your process and define/update your specifications.



You may also need to write some job descriptions.

Working method

You have to explain how you work. Some typical questions a rookie could ask you are:

- ✓ do you use github or gitlab?
- ✓ should I pull or merge? when? why?
- ✓ are we using kanban from github with CI/CD integration?
- ✓ what development process have you decided to use with what tools?

Quality assurance

How can you follow the quality of your project through time?

Define your use cases. Define your test scenarios. Write your test processes (automatic or manual).



You probably already have some user stories you wrote before, haven't you?

A document may define your different tests scenarios and their acceptance criteria. Some other documents may complete the tests campaign according to the scheduler defined in your project plan.

