

Service-oriented Architecture Lab

- Authors: L.Gaillard, M.Chevalier, P.Collet

Objectives

- Build and maintain a working ecosystem of microservices (working each week)
- Integrate micro-services using an Event-driven Architecture.
- Deliver and deploy a reproducible environment with Docker.
- Plan for and implement future requirements and features around your architecture.
- Identify the (very-)Minimal Viable Product that fulfills the requirements.

Project Vision: Blue Galactic X, A new space era

"I think we're going to the moon because it's in the nature of the human being to face challenges. It's by the nature of his deep inner soul... we're required to do these things just as salmon swim upstream." - Neil Armstrong

The third millennia is one of innovation, and one amongst many is the access to space which is overlooking many enhancements. Blue Galactic X is looking towards the future, implementing a system where they lift the barrier usually called the atmosphere, and strives towards making Humans a multiplanetary species. Obviously, there are many challenges down the road, but a wise man said that it is in the nature of the human being to aim for hard objectives, and not lie in complacency (*"We choose to go to the moon not because it's easy, but because it's hard."* - John F. Kennedy).

Streamlining the processes towards successful launches and successful missions is extremely important, and it involves many, many, many different actors and stakeholders. Organizing them into different services, while making this whole orchestra sound like a symphony is a challenge, it is *your* challenge.

Expected work (groups of 4)

Your role is to define, implement the *Minimal and Viable Product* (MVP) associated with the Blue Galactic X ambitions. The ambitions are going to be defined by a set of user stories. It is your duty to **identify the services** to expose in order to support these scenarios and to **properly integrate** them altogether in a viable way (e.g., persistent data, automated acceptance scenarios, controllable).

It is absolutely essential to maintain during the entire course of the project a **working** set of services, and which can prove that your entire system is functional: **space is expensive, failure is costly**. From the first set of user stories provided below, we will extend the supported features with new user stories, it should be part of the plan, while not overengineering your current implementation.

We insist, but keep in mind that the expected delivery is small in terms of business coverage and algorithmic logic, but it requires a non-negligible effort to **deliver it properly** (e.g., justified API and choices, persistence layer, turn-key containers).

Deliverables

- A github project belonging to the Polytech organization. Your project **MUST** comply to the following rules, assuming your team code is **x**:
 - The name should be *box-20-21-team-x*
 - There shouldn't be any git submodule
 - That's it for the first iterations, no particular structure inside the repository is demanded *yet*, as we just want to have a glimpse on what will be no more than a MVP. You will be the ones showing the instructors your scenarios, and you will just need to prove that your system supports the stories.
- The requirements will later be way stricter, and you will need to provide into your repository some scripts to install and run your project. More on that later, focus on building your first set of services and integrate them. For your information, every service will need to be dockerized in the near future.
- **A demonstrably working project needs to be ready to be shown each week.**
- A PDF report will later be included as well. Recording your architecture choices on a log somewhere could help you build your future reports.

Personas

- **Richard**, the Mission Commander, overlooking the whole mission.
- **Elon**, the Chief of the Rocket Department, overlooking the rocket itself and its systems
- **Tory**, the Launch Weather Officer, making sure the weather is safe across sites
- **Jeff**, the Telemetry Officer, handling the data communications between the flight hardware and the ground
- **Gwynne**, the Chief of the Payload Department, responsible of the customer's cargo and the efficient trajectory or orbit insertion

First set of user stories

1. As Tory (Launch Weather Officer), I need to check the weather status, so that I can be sure that the conditions are in a valid range for a safe operation of the rocket.
2. As Elon (Chief Rocket Department), I need to monitor the status of the rocket, so that I can be sure that the rocket is behaving correctly before launch.
3. As Richard (Mission Commander), I have to perform a Go/No Go poll to every monitoring department before giving the final go ahead with the launch, so that I can be sure that everything is nominal before launch.
The Go/No Go poll has to be done in the following order:
 - Weather Department (Tory)
 - Rocket Department (Elon)
 - Mission Commander (Richard)

4. As Elon (Chief Rocket Department), I have to send the launch order to the rocket after the GO from Richard, so that the rocket can launch into space and deliver the payload.

Added 28/09:

5. As Jeff (Telemetry Officer), I want to receive, store and consult the telemetry data of the rocket of the whole launch sequence (from before the launch, to the end of the mission), so that I can monitor that everything is working as intended, and that if anything goes wrong, I can find the root cause of the anomaly.
6. As Elon (Chief Rocket Department), I want to stage the rocket mid-flight (separate the rocket in 2 parts), so that the rocket remains as efficient as possible, by leaving behind the first stage, now empty in fuel, and continuing with the second stage and the payload, full in fuel.
7. As Gwynne (Chief Payload Department), I want to deliver the payload (satellite/probe) in space on the right orbit or trajectory, so that the customer's desires have been successfully fulfilled by the mission.

