# Elective exercise using Go and RPC

**Corso di Sistemi Distribuiti e Cloud Computing**
A.A. 2023/24

Valeria Cardellini

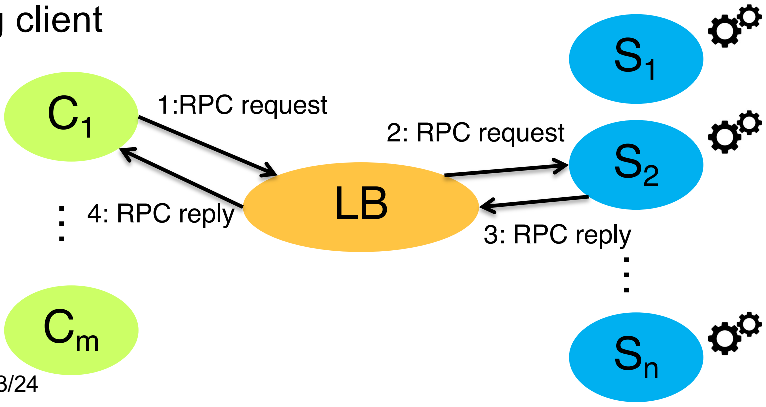
Laurea Magistrale in Ingegneria Informatica

## Elective exercise using Go and RPC

- Realize a mechanism to support in RPC the transparency to replication through a server-side load balancer that acts as a proxy between the clients and the replicated servers that provide the RPC service
- Requirements:
  - Use either Go and RPC or Go and gRPC
  - Organize properly your code into separate files
  - 1 student per team (2 students only if also the optional part is implemented)

# Overview

- The load balancer acts as a server towards the clients $C_1$, … $C_m$ and appears to offer them the RPC service
- The load balancer acts as a client towards the server $S_1$, … $S_n$
- For each client request, the load balancer selects one of the replicated servers (e.g., $S_2$) by means of a load balancing policy and sends it the client request
  - Policy: a simple one, either random or round-robin (preferred)
- The selected server executes the requested service and replies to the load balancer; in its turn, the load balancer replies to the requesting client



# Some simplifying assumptions

- The load balancer and the servers do not fail during computation
- The set of replicated servers is known to the load balancer when the system starts and is defined into a configuration file
- The ports are defined into a configuration file
- Choose your favorite RPC stateless service (not the same provided during the course!)

# Optional

- You can containerized your distributed application
  - To build the image, see Go official image hub.docker.com/_/golang
  - A Docker container per application component and then use Docker Compose to orchestrate the multiple containers on your computer

# Delivery

- When
  - By January 26, 2024
- What
  - Your code, including a README with instructions to run it
  - Optional: very short report describing the architecture of the distributed application and its main features
- How
  - By email
  - Use as mail subject: [SDCC] consegna esercizio in Go