

# Objective

Design and implement a **Layer-7 HTTP Load Balancer** in Go that:

- Accepts incoming HTTP requests from clients
  - Forwards them to healthy backend servers using configurable load-balancing strategies
  - Supports health checks, logging, and dynamic backend registration
- 



## Key Features



### Core

- Accept HTTP requests from clients (reverse proxy style)
- Route requests to one of many backend servers
- Read and forward full HTTP requests/responses



### Load Balancing Algorithms

- Round Robin
- Least Connections
- Random
- (Optional) IP Hash



### Health Checking

- Periodic health checks (HTTP ping) to each backend
- Mark servers as UP/DOWN

- Route traffic only to healthy backends

### **Logging**

- Log requests, target server, response time, and status code
- Optional: log to a file or external log service

### **Admin API (Optional)**

- Add/remove backend servers at runtime via REST API
- Check backend status (health + connections)

## **Deliverables**

- Complete Go codebase with README
- Config file and scripts to spin up mock backends
- Screenshots/logs showing request routing
- Optional: performance benchmark comparisons

---

## **Success Criteria**

- Handles at least 1000 requests/sec with 3 backends
- Balanced request distribution
- Handles backend downtime gracefully
- Clean and modular Go code with documentation