# Assignment 3: Message Queues

A common architectural pattern for high performance computing (HPC) tasks is using a queue in front of a service that performs some computational-intensive task. The queue receives requests containing a transaction and the calculation workers pull a request from the queue as soon as they are ready for processing. In this exercise, you will **develop your own message queue** to be used in combination with the ML prediction service that you will develop for the last assignment.

The message queue will be a service running on port 7500 of the host machine/container and expose the following operations via API:

- `push`: adds a new message to the queue and appends it to its end. The message should include all the data fields defined in Assignment 2 for the `transactions` or the `results` table in the transaction service.
- `pull`: removes the first (oldest) message from the queue and returns its contents to the caller.

There will also be an endpoint for listing, creating and deleting queues. When deleting a queue, all messages in the queue will be deleted as well. Only administrators will be allowed to delete or create queues. Append and pull will be allowed for agents and administrators only.

**Deadline:** 2025/05/19, 23:59 CET.

## Additional requirements

- The service will have a maximum number of messages per queue that can be configured using a configuration file.
- The service will gracefully handle any possible error condition (like queue non-existent, empty queue, queue too long, etc.).
- The service will write its queues to a persistent storage from time to time (time to persist can be configured using the configuration file). If the service is restarted, it will restore its state from the persistent storage.
- All files shall be submitted in a single `zip` file.
- A documentation file will also be included in the submission with a short description of the sumitted files.
- Every request performed by a client and all server responses must be logged with the following information: source, destination, headers (if applicable), metadata (if applicable), message body.

## Assessment

Total: 15 points.

- All requirements are satisfied: 10 points.
- The documentation is concise and technically correct: 5 points.