

Université Technique de Moldova

# RAPPORT

Au travail pratique Nr. 3 - 4  
A la discipline PAD

**Effectue:** st. gr FI-141  
**Verifie:** lect. superior

Cernei Eugeniu  
Alex Gavrișco

Chișinău 2018

## Laboratoire 3 : HTTP Protocol

### Tache :

Le tache a été de créer un serveur HTTP avec 2 ressources, et d'implémenter 3 opérations sur chaque ressource.

Les opérations minimaux sont « GET all », « GET by ID », « create ».  
Pour GET all il faut ajouter la possibilité de filtrer et paginer les données.

### La réalisation des taches:

Le code source pour l'application peut être trouvé dans le repository git:

**<https://github.com/wetosc/PAD-Labs/tree/master/lab3-4/code/dataNode>**

Pour ce projet j'ai utilisé la plateforme node.js, avec les libraires express et sequelize.

L'application a 2 ressources , Cellar et Wine, qui ont une relation « many-to-many » entre eux.

### Exemples des requêtes :

GET /api/wines/  
(GET all resources)

GET /api/wines/?offset=10&limit=5  
(GET all resources limited and with offset)

GET /api/wines/?qName=Purcari  
(GET all resources filtered by the param Name)

GET /api/wines/902320db-085a-4b3b-8ced-217cd08283f9  
(GET the resources with the given ID)

POST /api/wines/  
(Add a new resource described in the request body, application/x-www-form-urlencoded)

Les mêmes opérations peuvent être effectuées avec la ressource « Cellar ».

## Laboratoire 4 : Reverse Proxy

Tache :

Le tache a été de créer un reverse proxy pour le data-node crée au laboratoire précédent. Aussi, il faut implémenter le « load-balancing » et « caching » pour les requêtes.

La réalisation des taches:

Le code source pour l'application peut etre trouve dans le repository git:

**<https://github.com/wetosc/PAD-Labs/tree/master/lab3-4/code/proxy>**

Le serveur proxy reçoit les requêtes au port 3000 et les renvoie aux nœuds, qui écoute aux ports 3001 et 3002.

Les requêtes GET sont mises en cache pour 10 secondes, et si on fait la même requête 2 fois, on lui renvoie du cache. Pour ça, j'ai utilisé la base de données Redis.

Aussi j'ai implémenté le mécanisme de load-balancing « Round Robin ».

## Conclusion:

Dans ces travaux de laboratoire j'ai étudié les librairies les plus populaires pour créer un serveur web HTTP. Aussi, j'ai appris qu'est-ce qu'un proxy et comment le réaliser. Je pense que ces deux laboratoires sont proches de la vie réelle, parce que c'est le travail d'un développeur full-stack.