

Université Technique de Moldova

RAPPORT

Au travail pratique Nr. 2
A la discipline PAD

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

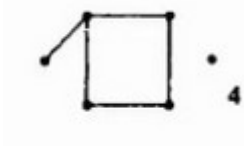
Cernei Eugeniu
Alex Gavrişco

Chişinău 2017

Laboratoire 2 : Distributed Data Collections

Tache :

Le tache a été de créer un collection des données distribue, avec un médiateur, avec la structure donnée.



L'idée générale est que le client demande des données au médiateur, qui fait le même requête a quelques nœuds centraux. Puis, on collecte récursivement les données et le médiateur les envoie au client.

Des autres taches ont été :

- D'implémenter différents formats de sérialisation des données (XML, JSON)
- De valider les données avec XSD ou JSON-schema
- D'offrir la possibilité au client de filtrer 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/lab2>

Pour ce projet j'ai créé deux librairies, tcpClient et eugddc. La première m'a aidé avec les connexions TCP, et la deuxième contient la logique commune pour autres modules de projet.

La communication d'entre les acteurs est homogène. Les messages ont la structure suivante :

```
type NodeMessage struct {
    Type      string
    Format    string
    Trace     []string
    Query     NodeQuery
    Data      []Dog
}

type NodeQuery struct {
    ID        string
    Query     string
    Params    *QueryParams
}

type QueryParams struct {
    Action    string
    Param     string
    Operation string
    Value     string
}
```

L'application a 3 acteurs :

- **Client**, qui fait une requête avec le Type = CLIENT a le mediator.
- **Mediator**, qui reçoit le message de client et fait une requête Type = GET a tous les nœuds connecte. Aussi, quand il reçoit des données suffisantes, il les transmis au client.
- **Node**, qui transmette les messages GET vers le bas, aux nœuds connectes, et les messages REPLY vers le haut. Il contient certain information. Les messages entres nœuds sont unidirectionnelles.

Presque toutes les paramétrés sont configurée de la ligne de commande, quand on lance l'acteur. On peut voir la liste des options avec la commande

```
./(insérer le nom d'acteur) -h
```

Aussi, j'ai crée un script qui lance toute les nœuds, mediator et client avec une seul commande :

```
./build.sh
```

Conclusion:

Dans ce travail de laboratoire j'ai étudié l'idée de collections des données distribue, la sérialisation des donnees et leur validation. Aussi j'ai implémente ces architectures, selon les taches de travail de laboratoire. Mémé si je ne connais pas ou on peut emploie ce type d'architecture, il a été une expérience intéressante.