

DOW

Titre : PFE2019#028 - Conception assistée d'IHM distribuées

Encadrant : Anne Marie Dery

Membres du groupe :

Valentin Ah-Kane - IHM

Baptiste Frère - AL

Alix Humbert - Web

I. Contexte & périmètre :

Le but de ce PFE est de créer une application Web destinée à aider les concepteurs d'applications à IHM distribuée. Ce sont des applications dont l'utilisation fait intervenir plusieurs interfaces, interactives ou non. Par exemple, un jeu de plateau pourrait être implémenté et une table tactile servirait de plateau tandis que les smartphones des différents joueurs seraient utilisés pour afficher leurs informations personnelles (ex : leur main de cartes). De manière générale, les différentes tâches impliquées dans l'utilisation de l'application à IHM distribuée sont réparties sur un ensemble de dispositifs prédéfinis. L'ensemble des tâches compose un arbre de tâches.

Nous reprenons le travail accompli par Simon Serrano et Valentin Ah-Kane durant leur stage de quatrième année. Notre travail consistera à améliorer les fonctionnalités déjà présentes et à en ajouter d'autres. A partir d'un arbre de tâches fournit par l'utilisateur, notre système doit lui permettre de :

- Visualiser les répartitions possibles des tâches sur les dispositifs disponibles (smartphone, tablette, table interactive, ...). Les répartitions sont calculées à partir d'un ensemble de règles. Par exemple une tâche privée, respectivement publique, sera attribuée à un dispositif privé (smartphone), respectivement publique (table interactive).
- Visualiser l'ensemble des façons de réaliser la tâche racine de l'arbre i.e les séquences d'actions possibles.

II. Défis identifiés :

- S'approprier et modifier le code tout en conservant les fonctionnalités déjà présentes.
- Changer l'architecture, décomposer en plusieurs services
- Introduire de la persistance
- Permettre à l'utilisateur d'ajouter de nouveaux dispositifs (ex : vidéoprojecteur)
- Changer le mapping entre les représentations RDF et Java de l'arbre des tâches. Le mapping actuel nécessite de modifier plusieurs classes pour ajouter de nouvelles propriétés aux tâches. Réduire le couplage entre le service Java chargé de la construction d'une version sérialisable de l'arbre et la bibliothèque Corese qui a connaissance d'une représentation objet de cet arbre.
- Mettre en place une notion de territoire sur des dispositifs sur lesquels on peut assigner des tâches
- Trouver des patterns pour la disposition de territoire sur les dispositifs en prenant par exemple en compte le nombre d'utilisateur
- Créer des nouvelles règles de répartition des tâches sur les dispositifs. Il n'y a actuellement que la règle donnée en exemple dans la partie contexte.

III. Activités déjà réalisées :

- Appropriation du fonctionnement actuel de l'application : lecture du code, review du code lors de réunions avec les membres actuels. Passation de projet effectuée par les initiateurs du projet.
- Identification des points critiques de l'architecture actuelle (monolithe)
- Conception de la nouvelle architecture à mettre en place (cf. annexe)

- Détermination des technologies qui vont être utilisées.
- Identification des besoins primaires. ex : bien que la persistance ne soit pas présente, elle n'est pas prioritaire sur l'ajoute de nouvelles fonctionnalités

IV. Listes d'activités prévues pour chaque semaine à plein temps

Semaine 1 : Début du développement du projet

- Refactor de l'architecture
 - Création des services imaginés (cf. annexe)
 - Extraction de l'ancienne base de code pour l'intégrer aux nouveaux services
 - Intégration des services (tests)
- Finir d'implémenter des fonctionnalités importantes non terminées pendant le stage concernant la modification du dispositif d'une tâche ou la duplication d'une tâche.
- Implémenter un premier exemple d'interface montrant un dispositif et son/ses territoires

Semaine 2 :

- Démonstration en début de semaine du travail réalisé
- Changer le mapping RDF/Java (cf. défis identifiés)
- Pouvoir ajouter une tâche à un certain territoire

Semaine 3 :

- Démonstration en début de semaine du travail réalisé
- Ajout de règles de répartition

Semaine 4 :

- Démonstration en début de semaine du travail réalisé
- Adapter la disposition des territoires proposés en fonction du nombre d'utilisateurs et des tâches associées
- Comparaison des règles de répartition (performance)

Semaine 5 : Finalisation du projet pour avoir un produit viable/utilisable et robuste.
Préparation des soutenances oraux et rédaction du rapport.

Organisation du travail (répartition de l'équipe)

Chaque membre va se focaliser sur la partie nécessitant les compétences de sa majeure. Valentin va principalement s'occuper de la partie frontend et de l'interaction avec l'utilisateur. Alix s'occupera principalement de la partie relative à RDF et SPARQL. Baptiste se consacrera aux aspects architecturaux. L'ensemble des membres veillera au bon fonctionnement de l'application, à la cohérence du système (interfaces des services, types de données échangées) et participera à le tester.

V. Annexes

