

Séance 1 - Etude de cas et modélisation

L'objectif de ce TD/TP est double: d'une part vous faire analyser et comprendre les enjeux de la mise en place de Lignes de Produits Logiciels (LPL) via l'étude d'un cas concret, et d'autre part vous initier à la modélisation de la variabilité, activité centrale dans le processus d'ingénierie des LPL. Les deux exercices étant indépendants, vous pouvez les traiter dans l'ordre que vous souhaitez. Il vous est demandé de produire un document récapitulatif répondant aux questions posées ci-après ainsi qu'un modèle de variabilité. Le travail est à effectuer en binôme, et sera à rendre selon les modalités suivantes:

- Document au format PDF + fichier *model.xml*, le tout dans une archive *NomBinome1_NomBinome2.zip*.
- A envoyer à clement.quinton@univ-lille.fr avant aujourd'hui (13/12/17) 16h30. **Il n'y aura aucune exception.**
- Objet du mail: [Rendu SPL1]

Exercice 1 : Analyse d'un cas d'étude

Lisez le rapport issu de la 12ème conférence internationale sur les LPL: "*HomeAway's Transition to Software Product Line Practice: Engineering and Business Results in 60 Days*" (oui, il est en anglais), que vous pouvez récupérer en cliquant ici. Répondez aux questions suivantes.

- Général

1. Quelle est l'offre de HomeAway? Quels sont ses clients?
2. Quel est son marché? Quelle est sa position dans le marché?
3. Quel type de logiciel est développé par HomeAway?

- Avant le "changement"

4. A l'origine, comment était développé le logiciel? Combien de "produits logiciel" avait HomeAway?
5. Quel type de variabilité trouvait-on et comment était-elle gérée?
6. Quelle a été la première étape pour mener à une solution unifiée? Qu'ont-ils utilisé?
7. Comparez la façon de travailler d'HomeAway avec la définition d'une LPL. Quels critères n'étaient pas remplis?

- Le changement

8. Quelle partie du cycle de développement a "crashé"? Quelles fonctionnalités ont montré les limites de l'approche de développement alors utilisée?
9. Qui a admis qu'il y avait besoin de changement?
10. Qu'a fait HomeAway au cours des 30 premiers jours de la transition?
11. Qu'a fait HomeAway durant les 30 jours d'après?
12. Quel rôle a joué BigLever/Gears?


- Après le changement

13. Comment HomeAway développe t-elle ses logiciels maintenant? Combien de "produits logiciels" ont-ils?
14. Quels sont les effets du changement sur les résultats des tests?
15. Quels sont les effets du changement sur la maintenance du code?
16. Qu'en est-il de la modélisation, de l'architecture ou de la documentation?
17. Comment a t'il été possible qu'HomeAway adopte une approche LPL en 60 jours? Habituellement, cela nécessite beaucoup plus de temps. Qu'est qui fait donc d'HomeAway un cas spécifique? #PROTip: pensez au type de logiciel qu'ils proposent. Quelle est la particularité?

Exercice 2 : Analyse

Il vous est demandé de modéliser la variabilité présente au sein d'une voiture en vous inspirant du texte suivant.

Lors de l'achat d'une voiture, le client se rend chez le concessionnaire qui l'aide à établir son choix en lui demandant de répondre à un ensemble de questions. Ces questions portent sur différents aspects de la voiture, tant matériels que logiciels/électroniques, et se répartissent en plusieurs catégories. Il y a notamment le type de voiture (citadine, break, coupé, SUV...), la finition (standard, sport, luxe), le nombre de portes, la motorisation, la motricité, etc. Le client peut également choisir les options qu'il souhaite. Ces options peuvent être des composants optionnels (*e.g.*, un toit ouvrant) ou des fonctionnalités optionnelles (*e.g.*, vitre arrière électrique).

1. Installez FeatureIDE. Rendez-vous sur la page de download et faites un drag-n-drop de l'icône  dans votre environnement Eclipse. Gardez les dépendances sélectionnées par défaut (d'autres sont encore en beta et viennent casser la configuration...) et confirmez.
2. Basculez en perspective FeatureIDE. Créez un nouveau projet FeatureIDE (laissez les paramètres par défaut, spécifiez un nom). Si ce n'est pas déjà fait, ouvrez *model.xml* avec le Feature Model Editor. Créez les fonctionnalités et les contraintes à l'aide de l'éditeur graphique. Vous pouvez également modifier le code du modèle via l'onglet *Source*, mais attention aux erreurs de syntaxe.

Quelques consignes:

- La liste des options donnée précédemment est bien entendu non exhaustive. Essayez de comprendre les dépendances parmi les options que vous modéliserez : quelles fonctionnalités sont obligatoires ? Quelles sont celles facultatives? Y-a t'il des alternatives? Quelles sont les fonctionnalités requises/exclues les unes des autres ?
- Soyez réaliste! Définissez des fonctionnalités et des dépendances qui ont du sens !
- Votre modèle doit être suffisamment fourni pour satisfaire un maximum de clients.