

Squelette possible de cahier des charges avec UML

Ceci est un exemple de squelette de cahier des charges avec les modèles UML vus en Travaux Dirigés.

Table des matières

Description Générale.....	2
Acteurs et Fonctions.....	2
Méthode Suggérée.....	2
Commentaires.....	2
Examen des Cas d'Utilisation.....	2
Élaboration du Diagramme.....	2
Commentaires.....	3
Arbitrages.....	3
Cas d'Utilisation :.....	4
Description Essentielle.....	4
Sommaire.....	4
Enchaînements.....	4
<i>Préconditions.....</i>	<i>4</i>
<i>Postconditions.....</i>	<i>4</i>
<i>Enchaînement Nominal.....</i>	<i>4</i>
<i>Enchaînements Alternatifs.....</i>	<i>5</i>
<i>Enchaînements d'Erreurs / Exceptionnels.....</i>	<i>5</i>
Commentaires.....	5
Description Réelle.....	6
Sommaire.....	6
Enchaînements.....	6
<i>Préconditions.....</i>	<i>6</i>
<i>Postconditions.....</i>	<i>6</i>
<i>Enchaînement Nominal.....</i>	<i>6</i>
<i>1.Enchaînements Alternatifs.....</i>	<i>6</i>
<i>Enchaînements d'Erreurs / Exceptionnels.....</i>	<i>7</i>
Commentaires.....	7
Prototypage / Maquettage.....	7
<i>Enchaînement Nominal.....</i>	<i>7</i>
<i>Enchaînements Alternatifs et enchaînements d'Erreurs / Exceptionnels.....</i>	<i>7</i>
Diagrammes d'Interaction.....	8
Diagrammes de Séquence « Système ».....	8
Enchaînement Nominal.....	8
Enchaînements Alternatifs et d'Erreurs / Exceptionnels.....	8
Diagrammes de Séquence et de Collaboration/Communication « Détaillés ».....	9
Références.....	10
Ateliers UML.....	10
Prototypage Rapide d'IHM.....	10
Dessin Vectoriel.....	10

Description Générale

Acteurs et Fonctions

Le premier pas consiste à analyser le problème pour trouver les « *acteurs* » du système à modéliser, c'est à dire les entités externes, humaines ou non, amenées à interagir avec lui.

Méthode Suggérée

Inventoriez dans une table les besoins fonctionnels du système, et pour cela dressez une liste de « *qui a besoin de faire quoi* ». Vous réserverez une colonne au centre pour indiquer si les acteurs ainsi identifiés sont *a priori* principaux (P) ou secondaires (S).

Acteur	Type	Besoin Fonctionnel
	-	
	-	
	-	
	-	

Le plus important dans cette étape est de bien couvrir la totalité des besoins fonctionnels du système à modéliser.



Conseils

Formulez vos fonctions de façon simple et concise, employez de préférence des verbes à l'infinitif. Restez également concret et gardez à l'esprit qu'il s'agit d'inventorier des fonctions destinées à être réalisées par un système.

Commentaires

Justifiez ici le fait que certains acteurs sont à coup sûr principaux car il ne fait aucun doute qu'ils devront interagir avec le système, c'est à dire qu'ils vont le manipuler pour réaliser une fonction dont ils attendront un résultat.

Justifiez à l'inverse le fait que d'autres acteurs seront secondaires car ils ne feront que recevoir de l'information.

Arbitrez enfin, en justifiant avec soin vos décisions, les situations moins tranchées. Pensez à rester au plus près du texte d'énoncé qui vous a été soumis.

Examen des Cas d'Utilisation

Élaboration du Diagramme

A partir de l'inventaire des fonctions établi précédemment, élaborer un diagramme UML des cas d'utilisation. Commencez par un diagramme de premier niveau sans chercher à trop détailler.

Figure: Diagramme des Cas d'Utilisation.

Une des difficultés de cet exercice est justement de ne pas tomber dans l'excès de détails.

Faites un diagramme simple ne dépassant pas un nombre raisonnable de cas d'utilisation (disons 10 juste pour fixer les idées), quitte à revenir plus tard sur les cas d'utilisation plus complexes dans un diagramme séparé. Il faut chercher à produire un diagramme efficace et ne pas vouloir être perfectionniste dès la première version car il y aura à itérer.

Conseils



Une bonne règle consiste à considérer qu'un cas d'utilisation doit concerner une seule fonction, un seul service, à exécuter par un seul acteur principal. Cette règle n'est pas toutefois une obligation absolue.

De plus, dans le cas de problèmes académiques (volontairement « simples et restreints », vous serez confronté à des problèmes faisant intervenir uniquement un ou deux acteurs et un ou deux cas d'utilisation.

Commentaires

Expliquez en quoi votre diagramme des cas d'utilisation est utile à la compréhension du problème que vous avez à traiter. Permet-il de mettre en lumière des groupes de fonctions, de les associer à des acteurs et/ou à des catégories de données ?

Arbitrages

Aucune solution n'est unique, et vous avez probablement arbitré entre plusieurs approches qui vous paraissaient toutes présenter leurs avantages et leurs inconvénients.

Commentez ici vos arbitrages les plus significatifs, ils permettront à votre lecteur de mieux appréhender le fil de votre approche du problème. Ces notes vous seront également utiles pour clarifier vos propres idées et maintenir une cohérence de vue tout au long du dossier.

Attention



Le fait que plusieurs solutions soient possibles n'implique pas qu'elles soient toutes équivalentes. En évoquant d'autres approches possibles, et les motifs de vos arbitrages, vous avez la possibilité de montrer en quoi votre solution est celle qui répond le mieux au problème posé.

Cas d'Utilisation :

Description Essentielle

Sommaire

Titre : Le titre du cas d'utilisation

Acteurs : Acteur1 (principal|secondaire), Acteur2 (principal|secondaire)

Résumé : Résumez en deux ou trois lignes en quoi consiste le cas d'utilisation. Vérifiez surtout que ce résumé reflète fidèlement ce qui est décrit dans les scénarios.

Fiche : Description Essentielle

Métadonnées :

Création :

Responsable :

Modifiée le :

Version :

Enchaînements

Préconditions

Pour l'exécution de ce cas d'utilisation, les conditions suivantes doivent être vérifiées au préalable :

1. ...
2. ...

Postconditions

Pour l'exécution de ce CU, les conditions suivantes doivent être vérifiées *a posteriori* :

1. ...
2. ...

Enchaînement Nominal

L'exécution nominale de ce cas d'utilisation s'effectuera selon le scénario suivant :

Acteur : ...	Système : ...
1. ...	
	2. ...
	3. ...
4. ...	
5. ...	

**Note**

Les étapes, ou les actions d'acteurs secondaires, se déroulant en dehors du système n'ont pas à figurer explicitement dans les enchaînements. Aussi, il vaut mieux éviter les phrases mettant un acteur secondaire comme sujet, car elles risquent de laisser à penser qu'il manipule le système.

Typiquement, l'exercice consiste à décomposer l'exécution du cas d'utilisation en étapes « atomiques » (c'est à dire non divisibles) dont on dresse une liste ordonnée dans le temps. Rédigez des phrases courtes et simples de la forme *sujet (acteur ou « système ») + verbe + complément d'objet*.

Enchaînements Alternatifs

A1 : Nom de l'enchaînement

4. point de départ sur l'enchaînement nominal
5. action spécifique
6. retour au point X de l'enchaînement nominal

A2 : Nom de l'enchaînement

8. point de départ sur l'enchaînement nominal
9. etc.

**Note**

Les enchaînements alternatifs et exceptionnels sont aussi à présenter sous forme de tableau.

Enchaînements d'Erreurs / Exceptionnels

E1 : Nature de la défaillance

15. point de départ sur l'enchaînement nominal
16. action spécifique
17. retour au point X de l'enchaînement nominal

Commentaires

Justifiez ici vos décisions relatives aux divers enchaînements alternatifs et d'exception, en expliquant les critères qui les ont guidés.

**Aide**

Quelle différence faut-il voir entre enchaînements *alternatifs* et *d'exception* (appelés aussi « *enchaînements d'erreurs* ») ? Une façon de les distinguer est de considérer que les enchaînements alternatifs résultent de *choix*, peu fréquents mais *voulus*, alors que les enchaînements d'erreur sont dus à une *défaillance* du système ou à une *mauvaise manipulation* d'un acteur.

**Conseils**

Par manque de temps, il est tentant de ne pas détailler les enchaînements alternatifs et-ou d'erreurs/exceptionnels. Ceci est fortement déconseillé car quel sera le comportement que programmeront (ou pas) les développeurs dans de telles situations. Aussi nous nous recommandons vivement d'identifier les enchaînements alternatifs et-ou d'erreurs/exceptionnels (même si les scénarios ne sont pas précisément détaillés).

Description Réelle

Sommaire

Titre : *Se reporter à la fiche de description essentielle*

Acteurs : *Se reporter à la fiche de description essentielle*

Résumé :

Se reporter à la fiche de description essentielle.

Fiche : Description Réelle

Métadonnées :

Création :

Responsable :

Modifiée le :

Version :

Enchaînements

Préconditions

Se reporter à la fiche de description essentielle.

Postconditions

Se reporter à la fiche de description essentielle.

Enchaînement Nominal

La description réelle de l'enchaînement reprend les étapes de la description essentielle en montrant, sur le(s) prototype(s) d'interface de la section suivante *comment* l'acteur va les exécuter. Il faut donc faire attention à ce que la description du scénario réel reste cohérente avec celle de l'enchaînement essentiel d'une part, et avec le prototype d'IHM d'autre part.

L'exécution nominale du cas d'utilisation se déroule selon les étapes suivantes :

Acteur : ...	Système : ...	Maquettes
1. ...		Maquette 1
	2. ...	
	3. ...	Maquette 2
4. ...		Maquette 3
5. ...		

1. Enchaînements Alternatifs

A1 : Nom de l'enchaînement

...

A2 : Nom de l'enchaînement

...

Enchaînements d'Erreurs / Exceptionnels

E1 : Nature de la défaillance

...



Note

Les enchaînements alternatifs et exceptionnels sont aussi à présenter sous forme de tableau.

Commentaires

Comme pour la description essentielle, justifiez ici vos arbitrages concernant les différents enchaînements alternatifs et d'exception, en expliquant les critères qui les ont guidés.

Prototypage / Maquettage

Enchaînement Nominal

Les maquettes ci-dessous permettent d'illustrer l'enchaînement nominal du cas d'utilisation qui nous occupe :

Figure : Maquette(s) d'interface pour l'exécution de l'enchaînement nominal.

La difficulté de l'exercice est de veiller à conserver la cohérence entre la description réelle de l'enchaînement nominal et cette(ces) maquette(s) d'interface. En particulier, tout changement d'un côté doit entraîner une vérification, et le cas échéant une actualisation, de l'autre.

Enchaînements Alternatifs et enchaînements d'Erreurs / Exceptionnels

Décrivez ici les interfaces nécessaires à la réalisation des enchaînements alternatifs, au moins pour les plus importants, ou pour ceux qui appellent une discussion.



Note

Il existe de nombreux outils de prototypage/maquettage rapide pour les maquettes d'interfaces, bon nombre d'entre eux libres et gratuits (voir références en fin de ce document). Pour les stades précoces des dossiers, préférez ceux donnant un aspect type « main levée », ils renforcent l'idée d'une maquette susceptible d'évoluer.



Aide

Si vous faites les efforts de bien détailler les messages et d'associer clairement les maquettes (une présentation par tableau telle que conseillée ci-dessus vous y aidera), le retour sur investissement vous paraîtra de façon plus claire. Ainsi, si un message est associé à plusieurs maquettes, il est probable que le message ne soit pas suffisamment détaillé. Si une seule maquette est associée à de nombreux messages, il est probable qu'il manque de maquettes.

Diagrammes d'Interaction

Il s'agit ici de présenter de façon graphique les scénarios précédemment établis. Ceci relève davantage du cahier des charges technique.

Par exemple, un processus de développement possible consiste à représenter chaque scénario par un diagramme de séquence « système » où le système est vu comme une boîte noire dont vous n'avez pas à détailler le fonctionnement interne. Il s'agit ensuite de détailler chaque diagramme de séquence « système » en un diagramme de séquence (détaillé) ou bien en un diagramme de collaboration/communication (détaillé) selon que le temps soit une notion importante à expliciter ou pas.

Diagrammes de Séquence « Système »

Enchaînement Nominal

Présentez ici le diagramme de séquence « système » représentant l'enchaînement nominal discuté dans la fiche de description essentielle du cas d'utilisation. Dans un diagramme de niveau « système », le système est vu comme une boîte noire dont vous n'avez pas à détailler le fonctionnement interne.

Figure: Diagramme de Séquence « Système » (DSS) pour l'enchaînement nominal.

Commentez ici les points remarquables de votre diagramme, et ce qu'ils sous-tendent en termes techniques.

Enchaînements Alternatifs et d'Erreurs / Exceptionnels

Présentez ici le(s) diagramme(s) de séquence « système » associés aux enchaînements alternatifs et d'erreurs discutés dans la fiche de description essentielle du cas d'utilisation.



Note

Dans la pratique, tous les scénarios (alternatifs, erreurs) du cas d'utilisation ne feront pas l'objet d'un diagramme de séquence « système ». Il appartient au responsable du dossier de conception de juger de l'intérêt de les réaliser.



Aide

Si vous avez fait l'effort de « bien » détailler complètement tous les messages échangés dans chaque scénario ainsi que toutes les informations contenues dans chaque message, alors le diagramme de séquence « système » devient une « simple » traduction graphique du scénario et son élaboration est relativement « facile ».



Conseils

Il est probable que des messages récursifs (du système vers lui-même) apparaissent. Cela correspondra certainement à des envois de messages entre les objets composant ce système ; messages et objets qui seront « confirmés » dans les diagrammes de séquence et/ou de collaboration/communication ultérieurs.

Diagrammes de Séquence et de Collaboration/Communication « Détaillés »

Détaillez ici les diagrammes de séquence « système » afin que le système soit vu comme une boîte claire.



Note

Dans la pratique, tous les diagrammes de séquence « système » ne sont pas détaillés. Il appartient au responsable du dossier de conception de juger de l'intérêt de les réaliser.

De plus il est possible de détailler les diagrammes de séquence « système » en utilisant des diagrammes de collaboration/communication si la notion de temps ne doit pas être explicite.



Conseils

Il y a plusieurs niveaux possibles dans la conception, ce qui renforce aussi l'idée de processus de développement itératif et incrémental tel que préconisé par/pour UML. A bas niveau, les objets en lien avec les diagrammes objets (puis diagrammes de classes) apparaîtront.

Références

De très nombreux outils existent pour élaborer des diagrammes UML comme pour réaliser rapidement des maquettes d'interfaces. Le support de cours du module fournit une liste assez complète dont voici un extrait.

Ateliers UML

Ateliers autonomes ou à intégrer dans des environnements de développement :

- **ArgoUML** : Libre, simple de prise en main, génération de code (Java, SQL...), module français, documentation abondante, ... - <http://argouml.tigris.org>
- **Modelio** : Outil professionnel passé récemment sous une licence « open source » - <http://www.modelio.org>
- **BoUML** : Outil libre de qualité utilisé il y a quelques années à l'IUT et devenu (faiblement) payant - <http://bouml.free.fr>
- **PapyrusUML** : Outil libre, professionnel, développé, maintenu et distribué par le CEA (Commissariat à l'Énergie Atomique) - <http://www.papyrusuml.org/>

Prototypage Rapide d'IHM

Logiciels permettant de réaliser rapidement des maquettes d'IHM :

- **Balsamiq Mockup** : Version complète installée à l'IUT, et en version d'essai gratuite pour 7 jours - www.balsamiq.com
- **Pencil** : Outil libre, existe sous forme d'addon de Firefox, et aussi en version autonome pour Windows - <http://pencil.evolus.vn>

Dessin Vectoriel

Outils de *dessin* uniquement n'offrant aucune fonctionnalité d'ingénierie :

- **Draw** de la suite bureautique libre LibreOffice (ou alternativement OpenOffice) : Intégré à la suite, mais à vocation généraliste (pas de palette UML) - www.libreoffice.org
- **Cacoo** : Outil web de création de diagrammes, croquis et schémas dont il existe une version gratuite « limitée » - <https://cacoo.com/>
- **Dia** : Outil libre de dessin vectoriel doté de nombreuses palettes (UML, réseaux,...) - <http://live.gnome.org/Dia> et aussi <http://projects.gnome.org/dia> (ancienne page)