

Projet de Développement Objet et Génie Logiciel

Fabien Picarougne, Antoine Pigeau, Julien Cohen, Marcus Barkowsky

Objectif

L'objectif de ce projet est de vous sensibiliser à l'ingénierie du logiciel dans un cadre de programmation orientée objet. Vous réaliserez une application en groupe, en suivant les différentes étapes d'un projet de développement informatique, et en adoptant une démarche objet pour les étapes de modélisation/conception et de développement. Chaque étape doit faire l'objet d'une fiche à retourner à l'encadrant. Le modèle de développement suivi est un modèle classique dit en V.

Sujets

Différents sujets sont proposés par les encadrants. La liste des sujets sera disponible le 13 février.

Contraintes

Ce projet est à réaliser par groupe de 4 étudiants. Chaque groupe sera suivi par l'un des quatre enseignants. Cet enseignant jouera le rôle d'un client qui commande un logiciel, et vous jouerez le rôle des employés d'une entreprise qui s'engage à fournir le logiciel.

Des fiches sont à livrer sur Madoc (cours Informatique de base S6) au fil de l'avancement du projet à des dates fixées ou à fixer par vous même. Merci de bien vouloir également informer l'enseignant affecté à votre groupe par e-mail lorsque vous effectuez un dépôt.

Pour chaque groupe, deux réunions doivent avoir lieu avec l'enseignant affecté : la première pour définir le cahier des charges (fiche n°1) et la deuxième pour la livraison (fiche n°7).

Fiche n°1 : lancement du projet, définition des besoins

Après avoir formé un groupe de 4 personnes, vous devez choisir dans votre groupe un coordinateur. Le coordinateur est l'étudiant chargé de déposer sur Madoc les fiches et autres documents associés, ainsi que d'assurer la communication avec l'enseignant. Il n'est pas un chef de projet (il ne prend pas de décisions), mais il doit s'assurer que les différentes étapes seront réalisées dans les délais annoncés.

Avant de déposer cette fiche, vous devez avoir pris rendez-vous avec votre « client » (l'enseignant). La rencontre avec le client doit servir à recueillir ses besoins et ainsi à permettre la rédaction du cahier des charges. Attention, le client peut énoncer des besoins ambigus ou incomplets, à vous d'interroger le client de manière à rendre le cahier des charges le plus précis possible. Le cahier des charges est contractuel : il établit ce que vous vous engagez à fournir.

Fiche à rendre avant le vendredi **16 février 2012**. Cette fiche devra contenir :

1. Noms des étudiants du groupe.
2. Nom et adresse e-mail de l'étudiant coordinateur du groupe.

3. Sujet choisi (après résolution des éventuels conflits entre groupes).
4. Date et heure prévue pour la réunion de lancement avec le client (vous devez fixer avant le 16 février la date du rendez-vous, qui pourra avoir lieu après le 16 février).
5. Date prévue de rendu de la fiche 2 (cahier des charges). A vous de fixer cette date à votre convenance (tenez compte des autres projets à rendre, des examens ou des vacances pour décider d'une date, et ce pour toutes les dates du projet). Une fois cette date fixée, vous devrez la respecter. Attention à ne pas faire durer trop longtemps chaque phase, sinon vous risquez d'avoir à terminer le projet dans l'urgence, peut-être en même temps que d'autres projets ou que des examens.

Fiche n°2 : cahier des charges & procédure de test de recette

Après avoir rencontré le client (l'encadrant joue le rôle du client), vous devez rédiger le cahier des charges conformément au besoin du client et le lui soumettre pour validation. Vous utiliserez tous les outils à votre disposition pour rendre le cahier des charges le plus clair possible (notamment des diagrammes UML). Vous prendrez soin d'éviter les sous-entendus, les ambiguïtés, la subjectivité, l'invérifiabilité, les imprécisions dans le cahier des charges. Par ailleurs, un cahier des charges doit décrire ce que fait le logiciel (comportement externe), mais pas *comment* il le fait (description interne).

Vous devez également fournir une procédure de test recette. Le test de recette permet au client de vérifier lors de la livraison que le logiciel fourni correspond bien à ce qui a été décrit dans le cahier des charges. Cette procédure doit décrire comment les tests seront effectués à la livraison (préciser par exemple si le logiciel est lancé sur le système du client ou sur celui du fournisseur), quels tests seront effectués et quels sont les résultats attendus pour chaque test. Dans l'idéal, chaque engagement pris dans le cahier des charges fait l'objet d'au moins un test, et chaque test est accompagné d'un scénario qui explique comment lancer le test. N'hésitez pas à présenter ces tests sous forme de tableau.

Fiche à rendre à la date convenue dans la fiche précédente. Cette fiche devra contenir :

1. Noms des étudiants du groupe.
2. Date de rencontre avec le client.
3. Document à joindre (PDF) : cahier des charges.
4. Document à joindre (PDF) : procédure de test de recette.
5. Date prévue pour une réunion du groupe pour la conception générale du système.
6. Date prévue de rendu de la fiche 3 (conception générale).

Chacun des documents joints (format PDF) doit mentionner :

- les noms du ou des rédacteurs,
- les noms du ou des relecteurs (lecture croisée).

Cette dernière remarque est valable pour tous les documents livrés des fiches suivantes, ainsi que pour les fichiers de code source.

Fiche n°3 : conception objet

Une fois le logiciel spécifié (dans le cahier des charges), vous devez concevoir/modéliser votre logiciel (avant de l'implémenter). La conception générale vise notamment à décrire l'architecture du système. Elle devra contenir :

- Un découpage en modules (par exemple un module pour l'IHM, un module pour les entrées/sorties fichiers, un module pour la gestion interne des données, un module pour les traitements à effectuer, etc.) accompagné d'une description en français des modules. Chaque module doit pouvoir être développé en parallèle avec les autres modules.
- Un ou plusieurs diagrammes UML pour décrire chaque module au niveau structurel (diagramme d'architecture, de déploiement, de classes) ou comportemental (diagrammes de séquences, d'état, de flot de données ou de flot de contrôle). On pourra utiliser différents niveaux de granularité avec la notion de paquetage selon que le diagramme est destiné au groupe (interaction des modules) ou au développeur d'un module.
Vous prendrez soin d'appliquer les principes de la modélisation/programmation objet.
- Le langage de programmation choisi (C++ ou Java).

Fiche à rendre à la date prévue dans la fiche précédente. Cette fiche devra contenir :

1. Noms des étudiants du groupe.
2. Date effective de la réunion de la conception générale.
3. Noms des participants à cette réunion.
4. Document joint (PDF) : conception objet.
5. Date prévue de rendu de la fiche 4 (code source des modules).
6. Date prévue de livraison des modules pour pouvoir réaliser leur intégration.
7. Diagramme prévisionnel de la répartition des tâches (diagramme de Gantt) indiquant pour chaque tâche la personne ou le binôme affecté. On trouvera au moins les tâches suivantes : une tâche de développement/documentation par module, l'intégration des modules, les tests d'intégration et/ou de pré-livraison, la rédaction du manuel d'utilisation, la rédaction du manuel d'installation.

Fiche n°4 : implémentation des modules

La conception terminée, chaque personne (ou binôme) doit développer le module qui lui est attribué.

Chaque module étant développé par une personne différente, il est primordial d'avoir une conception de bonne qualité. En effet, la conception est la seule référence dont un développeur dispose pour connaître les autres modules avec lesquels son module doit interagir.

Notez qu'on peut ajouter une étape de conception détaillée entre la conception générale et le développement mais elle n'est pas demandée dans le cadre de ce projet.

Chaque module doit être accompagné d'une documentation technique. La documentation technique doit permettre à quelqu'un qui n'a pas écrit un module de pouvoir corriger une erreur dans ce module ou d'y ajouter une fonctionnalité. Vous pouvez éventuellement utiliser un outil de production de documentation comme Javadoc.

Fiche à rendre à la date prévue dans la fiche précédente. Cette fiche devra contenir :

1. Noms des étudiants du groupe.
2. Documents joints : code des modules accompagnés de leur documentation technique (archive tgz ou zip).
3. Date de rendu de la fiche n°5 (intégration).

Fiche n°5 : intégration des modules

Une fois chaque module livré, vous devez les assembler pour former le logiciel complet. Normalement il n'y a pas de travail lors de l'assemblage, mais il se peut qu'on découvre une incompatibilité parce que l'un des modules ne respecte pas les interfaces, c'est-à-dire que les fonctions fournies n'ont pas le type spécifié ou bien qu'il a utilisé une fonction d'un autre module avec un type différent de celui qui a été spécifié et implémenté.

Une fois cette intégration terminée, vous devez tester le logiciel et, si possible, corriger les défauts constatés avant de le livrer. L'exécution de la procédure de recette est le minimum à faire pour tester le logiciel avant de le livrer.

Vous devez également prendre rendez-vous avec le client pour la livraison.

Fiche à rendre à la date prévue dans la fiche précédente. Cette fiche devra contenir :

1. Noms des étudiants du groupe.
2. Document joint (archive zip ou tgz) : programme complet (prêt à installer).
3. Document joint (PDF) : résultats des tests de pré-livraison.
4. Date prévue pour la livraison (tests/démonstration/exposé) en présence du client.

Fiche n°6 : livraison et bilan

Lors de la livraison, la procédure de test de recette sera suivie par le client. Vous devez noter les résultats obtenus et les commentaires du client. Pour chaque défaut constaté, vous devrez rendre compte de la raison du défaut et de la possibilité de le corriger ou non.

De plus, l'encadrant pourra vous demander de réaliser un exposé (détail à voir avec chaque encadrant).

Fiche à rendre avant **3 juin 2012** (la livraison doit avoir eu lieu avant). Cette fiche devra contenir:

1. Noms des étudiants du groupe.
2. Date de la réunion de livraison et noms des personnes présentes.
3. Bilan sur la réunion de livraison.
4. Liste des problèmes identifiés dans le logiciel livré.
5. Document joint (PDF) : manuels (utilisateur, installation).
6. Document joint (PDF) : Diagramme de Gantt effectif avec le nom des personnes ayant réalisé chaque tâche.
7. Document joint (archive zip ou tgz) : l'ensemble des sources du programme.

Répartition du travail

Il est demandé que chaque étudiant d'un groupe participe à la phase de développement, ainsi qu'à au moins une des autres phases (rédaction des autres documents). À part cette contrainte, à vous de répartir le travail à votre convenance dans le groupe pour ce projet. Toutefois, cette répartition doit être clairement indiquée, notamment en indiquant pour chaque document (textuel ou code source) le ou les auteurs, ainsi que le ou les relecteurs. Idem pour l'étalement dans le temps qui est libre, mis à part les deux échéances fixées.