

Analyse (R3.03) – TD 1 : Gestion d'un projet informatique et cycle de vie

Halim Djerroud

révision 2.2

Objectifs et organisation

Objectifs. À l'issue de ce TD, vous devez être capables de :

- identifier les activités du génie logiciel dans un projet réel, avec leurs entrées et leurs résultats ;
- reconnaître le cycle de vie suivi par une équipe et en analyser les limites ;
- choisir et justifier un cycle de vie ou une méthode (cascade, V, incrémental, Scrum, XP, Kanban, PU) adapté à un contexte ;
- réorganiser un projet de manière incrémentale (backlog, sprints, incréments).

Ce que vous devez faire

Vous allez lire l'histoire d'un projet étudiant, PRÊTMAT, qui a **échoué** : l'application a été rendue à temps, mais personne ne l'utilise. Votre travail consiste à jouer les **experts en gestion de projet** :

1. **Lire** l'étude de cas (contexte, compte rendu de réunion, journal de bord semaine par semaine).
2. **Analyser** ce qui s'est passé avec les notions du cours 1 : qui est le client, quelles activités ont été faites ou oubliées, quel cycle de vie l'équipe a suivi sans le savoir (exercices 1 à 3).
3. **S'entraîner** à choisir un cycle de vie sur sept autres projets (exercice 4).
4. **Refaire** le projet PRÊTMAT correctement, avec Scrum (exercice 5).

Toutes les réponses se trouvent dans l'étude de cas ou dans le cours : citez les faits du journal pour justifier.

Organisation. Travail en groupes de 3 ou 4. Les exercices 1, 2, 3 et 5 portent sur l'étude de cas PRÊTMAT ; l'exercice 4 est indépendant. Une mise en commun orale conclut la séance.

Étape	Thème	Durée indicative
Lecture	Étude de cas PRÊTMAT	15 min
Exercice 1	Identifier le projet	10 min
Exercice 2	Les activités du génie logiciel	25 min
Exercice 3	Organisation et cycle de vie	30 min
Exercice 4	Choisir un cycle de vie	40 min
Exercice 5	Rejouer le projet avec Scrum	40 min
Bilan	Mise en commun	20 min

Étude de cas : le projet PRÊTMAT

Contexte

Le département informatique d'un IUT prête du matériel à ses étudiants et à ses enseignants : cartes Raspberry Pi, kits Arduino (carte, câbles, capteurs), ordinateurs portables, caméras et casques de réalité virtuelle. Le parc compte environ 300 éléments.

Aujourd'hui, les prêts sont suivis par M. Petit, technicien du département, à l'aide d'un fichier tableur et d'une feuille papier signée au comptoir. Les problèmes sont récurrents : matériel perdu, retards non relancés, kits rendus incomplets, enseignants qui découvrent le jour du TP que les kits sont déjà prêtés. Les règles actuelles : un prêt dure 14 jours au maximum ; un enseignant peut bloquer plusieurs kits pour une séance de TP.

Mme Rousseau, cheffe de département, confie le développement d'une application web de gestion des prêts, baptisée PRÊTMAT, à une équipe de quatre étudiants de BUT2 dans le cadre d'un projet de 6 semaines. Un enseignant tuteur suit l'équipe et évalue le travail (dossier de conception en fin de semaine 2, soutenance en fin de semaine 6).

Contraintes imposées. Application web en PHP avec une base MySQL ; hébergement sur le serveur du département ; authentification par l'ENT de l'université (CAS) ; code versionné sous Git ; données personnelles des étudiants à protéger (RGPD). Les étudiants consacrent environ deux jours par semaine au projet, en parallèle de leurs autres enseignements.

L'équipe. Lina, Hugo, Karim et Emma.

Compte rendu de la réunion de lancement (semaine 1, lundi, 1 heure)

Présents : Mme Rousseau, l'équipe. **Absent :** M. Petit (en congé).

Besoins exprimés par Mme Rousseau :

1. consulter le matériel disponible ;
2. enregistrer un emprunt ;
3. enregistrer un retour ;
4. gérer le parc (ajouter, retirer un matériel) ;
5. se connecter avec son compte ENT ;
6. permettre aux étudiants de réserver un matériel en ligne ;
7. relancer automatiquement par mail les emprunteurs en retard ;
8. produire des statistiques d'utilisation ;
9. « on avait parlé d'une caution à un moment, à voir ».

Mme Rousseau précise qu'elle est très prise et qu'elle verra le résultat à la soutenance.

Journal de bord de l'équipe

Sem.	Ce qui s'est passé
S1	Réunion de lancement. L'équipe se répartit le travail par couche technique : Lina la base de données, Hugo l'interface, Karim le code serveur, Emma la documentation et les tests. Création du dépôt Git ; chacun travaille sur sa propre branche.
S2	Lina conçoit le modèle de données : un emprunt concerne un matériel. Hugo dessine les maquettes de 8 écrans. Emma rédige le dossier de conception à partir du compte rendu ; il est rendu au tuteur et bien noté.
S3	Lina crée la base. Hugo programme les pages avec des données fictives. Karim travaille sur la connexion à l'ENT (CAS). Emma commence un plan de tests.
S4	Karim termine l'authentification (une semaine et demie au total), puis code l'emprunt, le retour et la gestion d'une caution. Hugo continue les pages.
S5	Mardi : fusion des quatre branches, deux jours de conflits pour obtenir une version qui démarre. Vendredi : démonstration à M. Petit, revenu de congé. Il signale : (1) la plupart du matériel est prêté en kits , et il faut savoir quelles pièces manquent au retour ; (2) les enseignants réservent parfois 15 kits d'un coup pour un TP ; (3) il utilise une tablette au comptoir, or les pages ne s'affichent pas correctement dessus ; (4) il n'y a jamais eu de caution. Le modèle de données est à revoir.
S6	L'équipe adapte en urgence le modèle pour les kits. Emma teste l'application jeudi et vendredi : 11 bugs trouvés, 6 corrigés. Le rapport est rédigé la veille de la soutenance.

Soutenance et épilogue. La démonstration de l'emprunt fonctionne, mais le retour d'un kit incomplet provoque une erreur. La réservation en ligne, les relances et les statistiques n'ont pas été réalisées. Le tuteur salue la qualité du dossier de conception. Mme Rousseau demande pourquoi les réservations des enseignants n'apparaissent pas. L'application n'est jamais mise en service : M. Petit continue d'utiliser son tableur.

Exercice 1 : Identifier le projet (10 min)

1. Qui est le client du projet PRÊTMAT ? Qui sont les utilisateurs finaux ? Le client et les utilisateurs sont-ils les mêmes personnes ?
2. Quels sont les objectifs principaux du projet ? (3 à 5 objectifs)
3. Quelles sont les contraintes organisationnelles et techniques ?
4. Quelles sont les contraintes de temps (durée, jalons, livrables) ?
5. (*facultatif*) PRÊTMAT s'inscrit-il dans un **système d'information** au sens du cours ? Décrivez le SI existant avant le projet (ressources matérielles, logicielles, humaines, organisationnelles).

Exercice 2 : Les activités du génie logiciel (25 min)

Rappel

Les activités du génie logiciel vues en cours : (a) analyse des besoins, (b) spécification, (c) conception, (d) programmation, (e) validation et vérification, (f) intégration et gestion des configurations. Chaque activité a un **rôle**, des **entrées** et des **résultats**.

1. À partir du compte rendu et du journal de bord, complétez le tableau suivant pour le projet PRÊTMAT. Indiquez aussi ce qui a manqué.

Activité (cours)	Ce que l'équipe a fait (quand, qui)	Entrées utilisées	Résultat produit
Analyse des besoins			
Spécification			
Conception			
Programmation			
Validation / vérification			
Intégration / configurations			

2. Quelle différence faites-vous entre **spécification** et **conception** ? Illustrez chacune par un exemple tiré de PRÊTMAT.
3. Quelle différence faites-vous entre **validation** et **vérification** ? À quel moment chacune a-t-elle eu lieu dans le projet ?
4. Le dossier de conception a été bien noté, et pourtant l'application n'a jamais été utilisée. Comment l'expliquez-vous ?

Exercice 3 : Organisation et cycle de vie (30 min)

Rappel

Un **diagramme de Gantt** représente les tâches d'un projet dans le temps : une ligne par tâche, une barre horizontale allant de sa date de début à sa date de fin. Il fait apparaître les chevauchements, les dépendances et les jalons (livrables, soutenance).

1. Comment l'équipe s'est-elle répartie le travail ? Quelles difficultés cette répartition a-t-elle provoquées ?
2. Dessinez le diagramme de Gantt du projet PRÊTMAT tel qu'il s'est réellement déroulé (une colonne par semaine). Placez les jalons.
3. Proposez un schéma **sans le temps**, qui montre seulement l'enchaînement logique des activités.
4. À quel cycle de vie ce schéma correspond-il (cascade, V, incrémental, agile) ? Justifiez.
5. Parmi les limites de ce cycle vues en cours, lesquelles retrouvez-vous dans le journal de bord ? Citez les faits.
6. Quels objectifs ont été atteints ? Lesquels ne l'ont pas été ?
7. Quelles modifications de l'organisation auraient évité ces échecs ? Vers quel(s) cycle(s) de vie vous rapprochent-elles ?

Exercice 4 : Choisir un cycle de vie (40 min)

Pour chacun des projets suivants, choisissez le cycle de vie ou la méthode la plus adaptée parmi : **cascade**, **cycle en V**, **incrémental**, **Scrum**, **XP**, **Kanban**, **Processus Unifié (PU)**. Chaque réponse n'est utilisée qu'une fois. Justifiez votre choix en vous appuyant sur les caractéristiques du projet (stabilité des besoins, taille, criticité, implication du client, durée).

- Projet A. Pompe à insuline.** Une entreprise de matériel médical développe le logiciel embarqué d'une pompe à insuline. Les exigences sont définies par des normes et figées dès le départ. Le logiciel doit être certifié : chaque exigence doit être tracée jusqu'au test qui la vérifie.
- Projet B. Application de covoiturage étudiant.** Une jeune startup veut lancer une application mobile de covoiturage entre étudiants. Le fondateur a une idée générale mais ne sait pas exactement quelles fonctionnalités plairont. Il veut tester rapidement auprès de vrais utilisateurs et ajuster le produit selon leurs retours.
- Projet C. Support applicatif d'une mairie.** Une petite équipe assure la maintenance des logiciels d'une mairie. Elle reçoit en continu des demandes de correction et de petites évolutions, imprévisibles, dont les priorités changent d'un jour à l'autre.
- Projet D. Script d'export comptable.** Un développeur seul doit écrire, en deux semaines, un programme qui exporte les données comptables d'une base vers un format imposé par l'administration fiscale. Le format est parfaitement documenté et ne changera pas.
- Projet E. Logiciel de caisse pour une chaîne de boulangeries.** Le client veut pouvoir encaisser au plus vite dans ses magasins. La gestion de la carte de fidélité puis les statistiques de ventes pourront arriver ensuite, dans des versions ultérieures.
- Projet F. Refonte du SI des inscriptions d'une université.** Projet de deux ans, plusieurs équipes, nombreux acteurs (étudiants, scolarité, enseignants, services financiers), nombreux cas d'utilisation. L'architecture est critique (montée en charge lors des inscriptions, interfaçage avec des systèmes nationaux) et doit être validée tôt.
- Projet G. Moteur de tarification d'une assurance en ligne.** Une équipe de cinq développeurs, réunie dans le même bureau, maintient le code qui calcule le prix des contrats. Un actuair de l'entreprise travaille à plein temps avec l'équipe et modifie les règles de tarification presque chaque semaine. Une erreur de calcul coûte très cher : le code doit rester fiable et facile à modifier en permanence.

Questions complémentaires :

8. Pour le projet B, décrivez ce qui risquerait de se passer si l'on choisissait un cycle en cascade.
9. Pour le projet A, pourquoi une méthode agile « pure » poserait-elle problème ?
10. Un projet réel suit-il toujours un seul cycle de vie ? Donnez un exemple de combinaison.

Exercice 5 : Rejouer PRÊTMAT avec Scrum (40 min)

On donne une seconde chance à l'équipe : même sujet, mêmes contraintes, mêmes 6 semaines, mais cette fois avec Scrum et des sprints de 2 semaines.

Rappel

Le **backlog produit** est la liste ordonnée de tout ce que le produit doit faire, la plus prioritaire en haut. Il est tenu par le **Product Owner**. Chaque élément est souvent écrit sous forme de **récit utilisateur** (*user story*) :

« En tant que *<rôle>*, je veux *<action>* [afin de *<bénéfice>*]. »

Exemple : « En tant qu'enseignant, je veux consulter le matériel disponible afin de préparer mon TP. » À chaque sprint, l'équipe prend les éléments du haut du backlog et livre un **incrément** utilisable, montré au Product Owner en fin de sprint.

1. (*facultatif*) Donnez une définition simple de la **réussite** d'un projet. Selon cette définition, PRÊTMAT est-il une réussite ?
2. L'équipe voulait réaliser les 9 besoins de la réunion de lancement. Un projet qui comporte beaucoup de fonctionnalités a-t-il plus de chances de réussir ? Justifiez.
3. Quelle version minimale de PRÊTMAT, livrée tôt, aurait déjà été utile à M. Petit ?
4. Qui doit jouer le rôle de **Product Owner** ? Qui peut être **Scrum Master** ? Justifiez, en tenant compte des personnes décrites dans l'étude de cas.
5. Rédigez le **backlog produit** de PRÊTMAT : 6 à 8 éléments, formulés du point de vue d'un utilisateur, classés par priorité. Intégrez ce que l'on sait maintenant (kits, TP, tablette).
6. Répartissez le backlog en 3 sprints. Pour chaque sprint, indiquez l'**incrément** livrable et à qui il est montré.
7. Avec cette organisation, à quel moment le problème des kits aurait-il été découvert ? Qu'est-ce que cela aurait changé ?
8. (*facultatif*) Sous quelles conditions peut-on mettre en place un cycle incrémental ? Sont-elles réunies ici ?

Mise en commun (20 min)

- Chaque groupe présente en 2 minutes son Gantt de PRÊTMAT et la principale cause d'échec qu'il retient.
- La classe confronte ses réponses à l'exercice 4.
- **Et vous ?** Dans votre dernière SAÉ, retrouvez-vous certains symptômes de PRÊTMAT (utilisateur rencontré tard, intégration en fin de projet, tests la dernière semaine) ?