

Plan de développement

Version	1.0
Date	21 mai 2012
Rédigé par	Florian GUILBERT
Relu par	Giovanni HUET
Approuvé par	
Signature	

MISES À JOUR

Version	Date	Modifications réalisées
0.1	30/11/2011	Création
1.0	15/01/2012	Relecture par Giovanni HUET
1.1	29/03/2012	Modification de l'organigramme des tâches (log de connexion)
1.2	22/04/2012	Modification du planning

Table des matières

1	Contexte du projet	4
2	Documents applicables et de références	4
3	Terminologie et sigles utilisés	4
4	Méthodologie et développement	4
5	Organisation et responsabilités	6
6	Organigramme des tâches	7
7	Évaluation du projet et dimensionnement des moyens	7
8	Planning général	8
9	Procédés de gestion	8
9.1	Gestion de la documentation	8
9.2	Gestion des configurations	9
10	Revue et points clefs	9
11	Procédures de suivi d'avancement	9

1 Contexte du projet

Ce projet propose une solution d'enregistrement décentralisée pour le web. En effet, la gestion des identifiants/mots de passes et l'enregistrement auprès de service est un problème récurrent du web.

Notre solution consiste à mettre en place un site web qui permettra à ses utilisateurs de les authentifier auprès de lui et ensuite garantir leur authentification auprès de tous les autres sites, en admettant que ceux-ci proposent un service d'authentification externe. Ce site web se basera sur un protocole que nous mettrons au point, au préalable.

Dans cet analyse du projet, nous admettons que nous allons pouvoir trouver un protocole sécurisé répondant aux objectifs du client, car dans le cas contraire, il sera impossible de développer le logiciel. Il faudrait uniquement produire un document prouvant pourquoi un tel protocole est impossible à réaliser.

Ce projet a été proposé par M. Florent NICART dans le cadre de notre formation en Sécurité des Systèmes Informatiques à l'Université de Rouen.

Étant un projet universitaire, ce travail a pour but de nous apprendre à gérer un projet, de sa partie analyse jusqu'à sa complète réalisation. Pour cette même raison, le projet devra être impérativement terminé avant la fin du second semestre.

2 Documents applicables et de références

- Intitulé du sujet du projet ;
- spécification technique de besoin ;
- cahier des recettes ;
- document d'architecture logicielle ;
- analyse des risques.

3 Terminologie et sigles utilisés

AdR : Analyse des Risques ;

CdR : Cahier de Recettes ;

DAL : Document d'Architecture Logicielle ;

PdD : Plan de développement ;

STB : Spécification Technique de Besoin ;

SAUD : Site d'Authentification Unique et Décentralisé ;

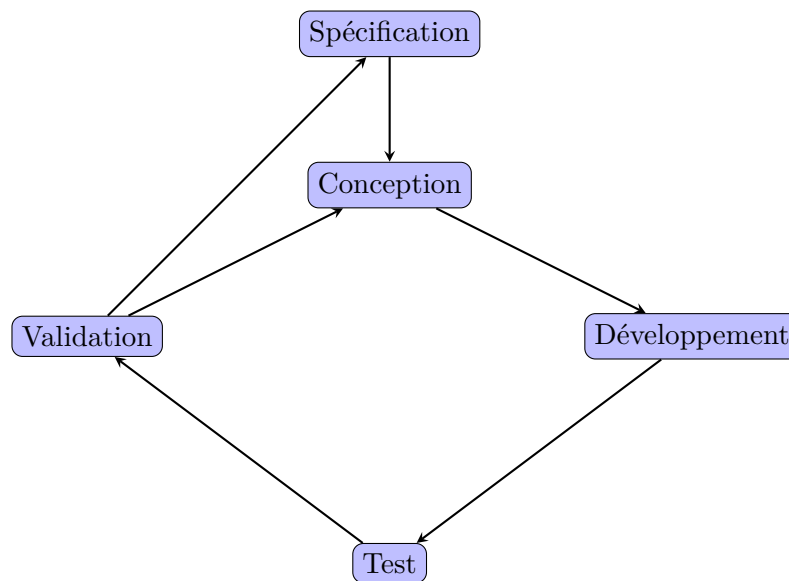
API : (Application Interface Programming), ensemble de fonctions et de modules permettant de réaliser un but précis ;

IHM : Interface homme-machine, (interface graphique) ;

RFC : (Requests for comments) est un document officiel décrivant les aspects techniques d'Internet ou de différents matériels informatiques (routeurs, serveur DHCP...) ;

4 Méthodologie et développement

Le développement du projet suit le schéma suivant :



Spécification

- Déterminer les objectifs ;
- définir les contraintes ;
- évaluer les risques.

Responsabilité : clients, testeurs et programmeurs.

Conception

- Définir les composants à développer.

Responsabilité : testeurs et programmeurs.

Développement

- Développement des composants ;
- tests unitaires sur ces composants.

Responsabilité : programmeurs.

Test

- Tester des scénarii de tests.

Responsabilité : testeurs.

Validation

- L'étape de validation détermine si la version développée propose bien les fonctionnalités attendues et dans ce cas valide la version du logiciel. Le travail pour la prochaine version peut commencer ;
- s'il n'y a pas validation de la version, alors c'est aux testeurs de décider s'il faut repasser l'étape de spécification ou s'il faut retourner directement à l'étape de conception ;
- dans le cas d'un succès d'une validation, une version du logiciel peut être livrée.

Responsabilité : clients et testeurs.

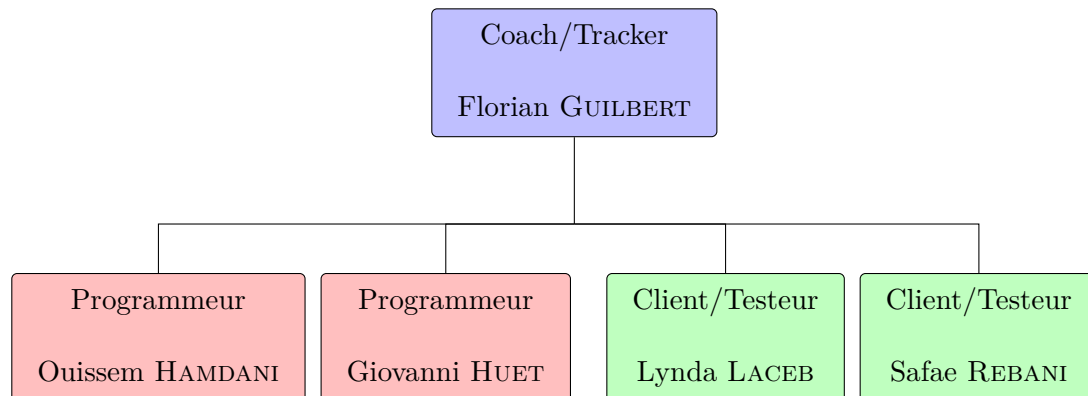
Cette méthodologie de développement suit la méthode XP (eXtreme Programming). Durant le développement de ce projet, il y aura trois itérations.

1. Conception théorique du protocole sécurisé entre un SAUD et un site quelconque ;
2. développement de la gestion utilisateur du SAUD (site d'authentification unique et décentralisée) ;

3. implantation du protocole, coté serveur (sur le SAUD) et développement d'une API permettant d'implanter le coté client du protocole sur un site quelconque.

Chacune des ces étapes comportent des modules qui seront développés en binôme afin d'une part, de faire en sorte que le plus de programmeurs possibles maitrisent le code source et d'autre part qu'une absence ne nuise pas au processus de développement.

5 Organisation et responsabilités



Coach :

- Garant du processus et de la méthodologie ;
- vérifie que chacun joue son rôle ;
- organise et anime les réunions et les séances de planifications ;
- valide les orientations techniques.

Client :

- Spécifie les fonctionnalités du logiciel ;
- communique les informations utiles aux développeurs et reçoit leurs “feedback” ;
- définit les fonctionnalités à partir de scénarii d'utilisations ;
- spécifie les tests de recette.

Programmeur :

- Responsables de la production du code ;
- conçoit pour assurer la pérennité du code ;
- teste pour assurer la qualité du code ;
- émet et révisé des estimations de charge.

Testeur :

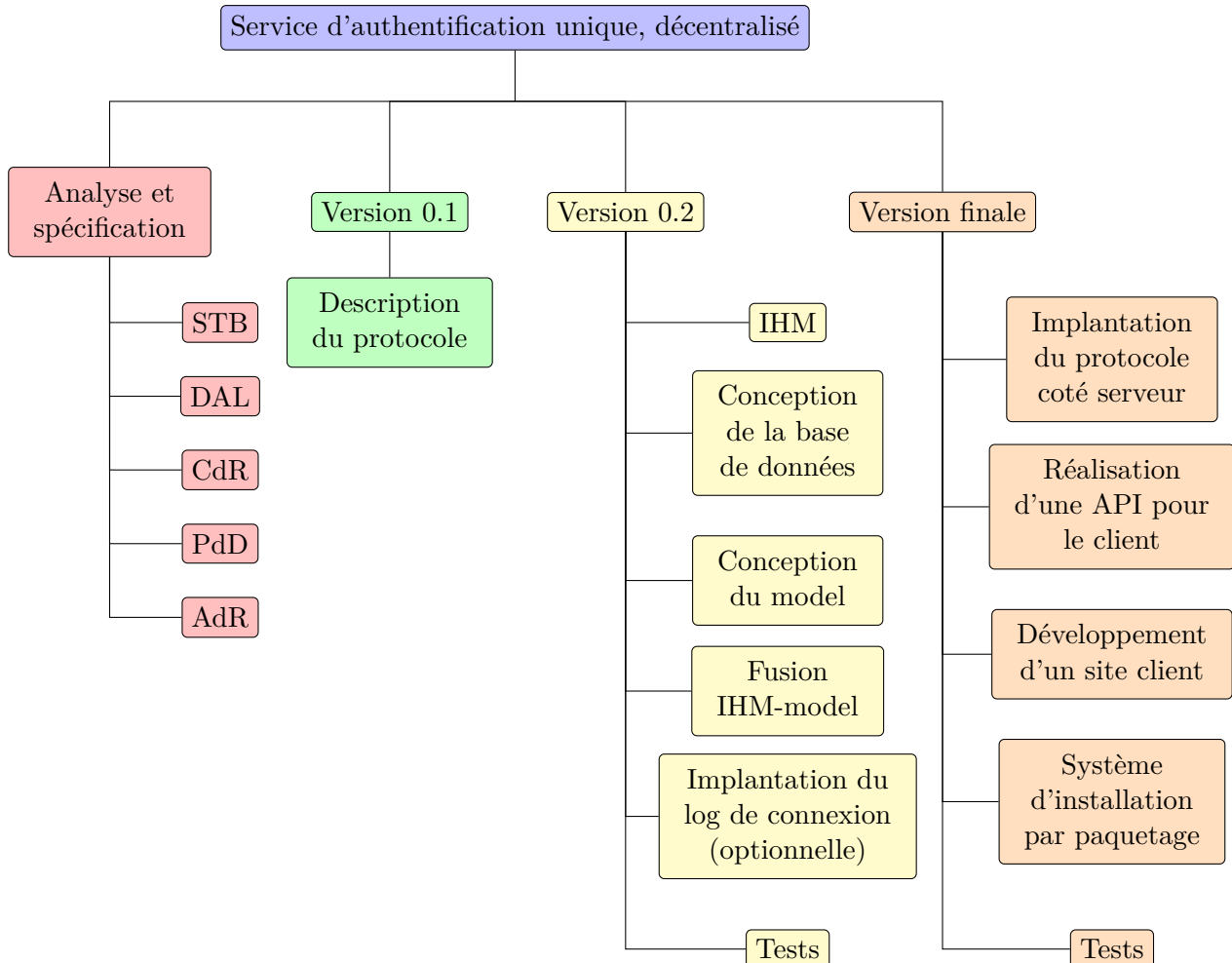
- Conçoit et réalise les tests de recettes définis par le client ;
- recherche l'automatisation des tests ;
- développe les outils de tests nécessaires et les scripts à exécuter ;
- témoigne de l'avancement du projet.

Tracker :

- Assure le suivi du planning ;
- cherche à détecter les difficultés le plus tôt possible et en informe le coach.

6 Organigramme des tâches

Le projet va être organisé en trois versions (une par itération), le schéma suivant décrit les tâches à développer pour chaque version :



7 Évaluation du projet et dimensionnement des moyens

Version 0.1 Cette première version est un peu particulière, car elle ne modifie en rien le logiciel. Il s'agit pour cette version d'établir un protocole permettant au produit de fonctionner de manière totalement sécurisée. C'est donc une partie intégralement théorique.

Aucun besoin de ressources ou de moyens.

Version 0.2 Le but de cette version est de poser les bases du logiciel. Par conséquent, il faut développer d'une part l'interface graphique afin de pouvoir tester le logiciel et d'autre part configurer la base de données afin de pouvoir, par la suite utiliser des jeux de données pour les tests.

Dans un second temps et pour pouvoir proposer une version fonctionnelle, il faudra implanter une gestion d'utilisateur dans le but qu'un utilisateur puisse se créer un compte et le gérer.

Pour le développement, les seuls besoins en moyens sont de travailler sur des machines disposant de serveur web Apache.

Pour le test, aucune ressource ni moyen requis, au contraire, pour tester la compatibilité du produit, il faudra tester avec le plus de technologies web différentes (différents navigateurs, systèmes d'exploitation, ...).

Version 1.0, finale Cette version ajoute à la version 0.2 l'implantation du protocole établie en 0.1.

Le contexte de développement et de tests sont similaires à celle de la version 0.2.

8 Planning général

Nom	Date de début	Date de fin	Ressources
* Élaboration du protocole (RFC)	23/01/12	17/02/12	Équipe [100%]
• Conception de l'interface graphique	20/02/12	24/02/12	Équipe [50%]
• Conception de la base de données	20/02/12	24/02/12	Équipe [50%]
• Réalisation du model	27/02/12	02/03/12	Équipe [100%]
• Fusion de l'interface graphique et du model	12/03/12	16/03/12	Équipe [100%]
• Développement - historique de connexion	19/03/12	23/03/12	Équipe [50%]
• Test de la gestion du compte	19/03/12	23/03/12	Équipe [50%]
• Implantation du protocole coté serveur sur le SAUD	26/03/12	20/04/12	Équipe [33%]
• Développement du SE et réalisation de l'API	26/03/12	20/04/12	Équipe [33%]
• Conception d'un paquetage d'installation	07/05/12	11/05/12	Équipe [33%]
• Test de l'application	07/05/12	11/05/12	Équipe [33%]
• Rédaction du protocole	26/03/12	12/05/12	Équipe [33%]

FIGURE 1 – Diagramme de Gant, tâches

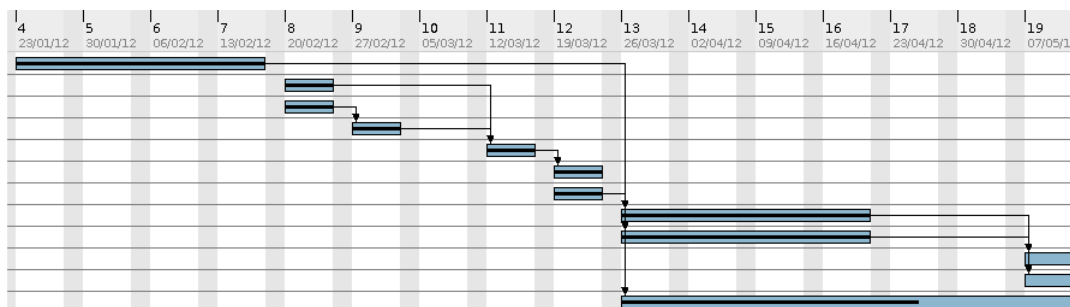


FIGURE 2 – Diagramme de Gant, graphes

Sur ce graphe, on peut observer trois couleurs, chaque couleur désigne une version du logiciel différente (rouge : 0.1, bleu-vert : 0.2, bleu : 1.0 (finale)).

9 Procédés de gestion

9.1 Gestion de la documentation

Il y aura deux documentations livrables :

- Le manuel d'utilisation rédigé par l'équipe testeur ;
- La RFC du protocole rédigé par l'équipe.

Les documents utilisés tout au long de la réalisation de ce projet sont :

- Analyse des risques (AdR) rédigé par Giovanni HUET et relu par Lynda LACEB
- Cahier des recettes (CdR) rédigé par Lynda LACEB et Safae REBANI et relu par Ouissem HAMDANI ;
- Document d'architecture logicielle (DAL) rédigé par Ouissem HAMDANI et relu par Safae REBANI ;
- Plan de développement (PdD) rédigé par Florian GUILBERT et relu par Giovanni HUET.
- Spécification technique des besoins (STB) rédigé par Lynda LACEB et Safae REBANI et relue par Florian GUILBERT ;

9.2 Gestion des configurations

Toutes les versions du programmes seront gérées par le logiciel de versionnage **Subversion** (SVN). Toutes les machines servant à développer devront avoir les même configurations des logiciels Apache, PHP et MySQL. Il n'y a pas de configurations particulières pour les machines visant à tester le logiciel hormis de disposer d'une connexion internet et d'un navigateur (à jour).

10 Revues et points clefs

Revue 1 : La première revue du projet aura lieu le 19 janvier 2012 lors d'une soutenance ayant pour but de présenter le projet aux responsables de la matière "Gestion de projet" et aux clients. Il est demandé auparavant d'envoyer aux examinateurs de cette soutenance une version de chaque document précités : (AdR, CdR, DAL, PdD, STB).

11 Procédures de suivi d'avancement

Le suivi d'avancement du projet sera effectué par l'intermédiaire de réunions régulières et de jalons. Ceux-ci permettront de contrôler l'état d'avancement de chaque module et de réagir rapidement à tout écart avec les prévisions en réorganisant le planning.

Il y aura par conséquent une réunion chaque semaine (hormis durant les vacances) entre les membres du groupe. À chaque validation, il y aura présentation au client du travail effectué.

Des réunions avec le client pourront aussi être organisées si le client en fait la demande ou bien qu'un membre du groupe souhaite bénéficier de son avis.

Pour chaque réunion, un compte-rendu sera réalisé et placé à disposition du groupe par l'intermédiaire de Subversion.