

LE CERCLE DES SCRIBOUILLARDS

un projet supervisé par Vincent Boutour



Mohamed Lebib

Bettina Azza

Sami Bouhaf

Marina Passoubady



Introduction

Notre projet porte sur la création d'un site web semblable à Reddit. Créé en 2005, Reddit est un site communautaire de partage de signets, il permet à des utilisateurs de soumettre leurs liens et de voter pour les liens proposés par les autres utilisateurs.

Notre site permet à des utilisateurs de proposer des articles et des thèmes. Un utilisateur doit posséder un compte pour accéder à nos services. Il peut proposer un thème et consulter ceux proposés par d'autres scribouillards. Tous les utilisateurs peuvent rédiger un article sur les thèmes proposés. De plus, les articles peuvent être valorisés à l'aide de votes (positif ou négatif), attribués par les scribouillards.

Notre site se nomme : « Le cercle des scribouillards ». Nous avons choisi ce nom en référence au film « Le cercle des poètes disparus ». Le terme scribouillard, un peu péjoratif, désigne les personnes travaillant dans l'administration. Nous avons volontairement associé ces deux références pour donner un côté décalé. En effet, nous avons réalisé un site internet ce qui n'est ni de l'époque du film ni de celle du terme scribouillard datant du 19^{ème} siècle. Nous voulions donner un nom littéraire au site bien que son utilité soit moderne et vise un public jeune.

Dans le cadre du module de génie logiciel, le projet doit respecter les consignes suivantes :

Le projet doit fonctionner sur tous les supports, nous devons utiliser git et Markdown. Le projet doit être réalisé en Java/Javascript ou Go et respecter le principe d'inversion de contrôle. Nous devons également utiliser un outil de gestion de dépendance tels que Maven, Gradle, Npm etc. De plus, nous devons effectuer des tests unitaires, rédiger les besoins, une documentation complète et refactoriser le code au fur et à mesure du cours.

Sommaire

Introduction	2
Mise en application.....	4
Organisation d'équipe et mise en place	4
Présentation du projet	5
Diagramme de cas d'utilisation	5
Diagramme de classes	7
Fonctionnalités	9
Tests unitaires	16
Tests fonctionnels	16
Difficultés rencontrées, solutions trouvées	17
Points perfectibles, restes à faire, points non solutionnés	18
Retours personnels sur le module.....	19
Retour globale de l'équipe des Scrib's	19
Retour personnel de Sami BOUHAFS	19
Retour personnel de Bettina Aza	19
Retour personnel de Marina Passoubady	20
Retour personnel de Mohamed LEBIB	20
Conclusion.....	21

Mise en application

Avant de concevoir et réaliser notre projet en génie logiciel, nous avons d'abord suivi des cours théoriques. Ces cours ont servi à nous exposer les différents outils informatiques que nous pouvions ou devons utiliser pour répondre aux besoins de notre projet.

Parmi les outils imposés, nous avons fait le choix d'utiliser STS et le langage de programmation JAVA. C'est le langage de programmation que l'ensemble du groupe maîtrise d'autant plus que nous l'avons étudié lors de nos études antérieures et en début de semestre.

En ce qui concerne le respect de l'inversion de contrôle, notre choix c'est automatiquement tourner vers Spring qui est en adéquation avec JAVA. Pour gérer les dépendances, nous utilisons MAVEN car c'est l'outil le plus utilisé pour les projets JAVA avec le Framework Spring.

De plus, certains membres de l'équipe l'ont déjà utilisé. Parmi les outils présentés en cours, GitHub est l'outil que l'on a jugé le plus adapté au travail d'équipe sur un projet. Au départ cette outil nous semblait difficile d'utilisation mais nous avons réussi à surmonter la difficulté à l'aide des cours et de tutoriels sur internet.

Organisation d'équipe et mise en place

Au début du projet, nous avons très vite décidé de nous séparer en 2 binômes. L'un d'entre eux est consacré à la partie développement et le second à la documentation. Cette organisation nous a paru évidente suite au domaine de prédilection de chacun.

En parallèle nous avons créé un projet git dans lequel nous déposons nos avancés au fur et à mesure. Pour avoir un suivi de l'avancement du projet, nous avons créé un tableau de bord sur l'outil Trello.

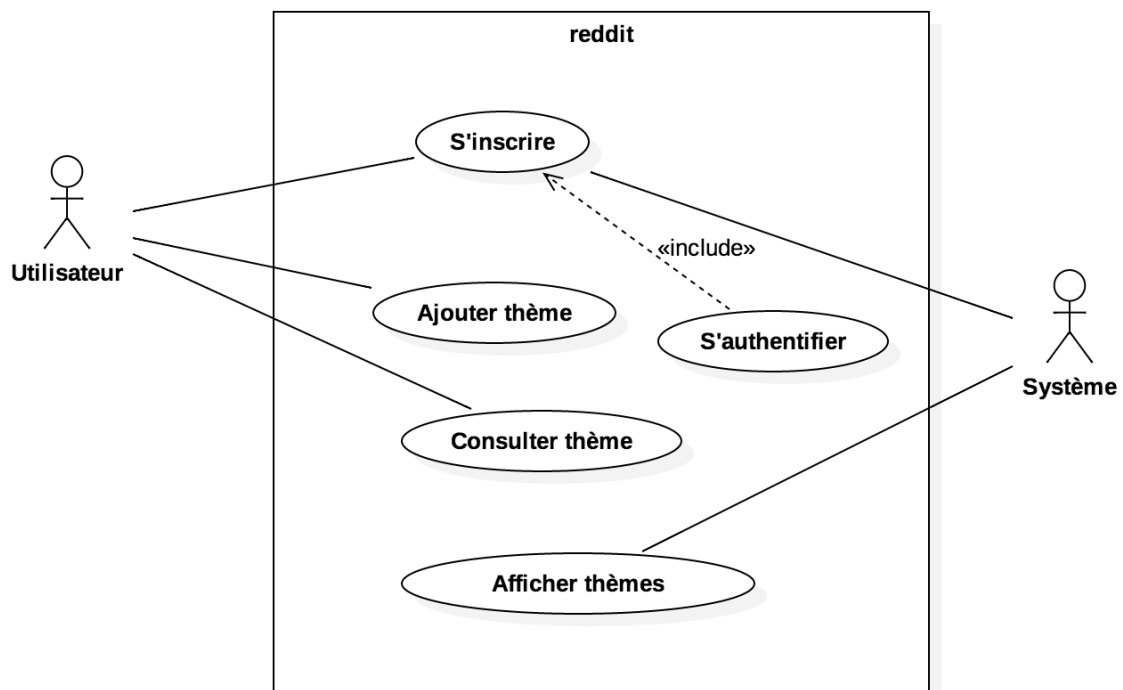
Pour coordonner les deux équipes, nous effectuons régulièrement des « team meeting » sur Skype ou lors des séances de cours. Nous nous sommes constamment consulté les uns et les autres pour rapporter des améliorations à nos tâches ou venir en aide à ceux qui en avaient besoin.

Présentation du projet

Les diagrammes UML nous ont permis de délimiter les fonctionnalités de notre projet et nous ont servis de base pour la réalisation de celui-ci. Nous avons choisi de réaliser uniquement les diagrammes de contexte de classes car ils étaient pour nous les plus pertinents.

Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation nous a servi de base pour la réalisation du projet car il en contient toutes les fonctionnalités. L'utilisateur a la possibilité d'effectuer trois différentes actions. En effet, il a la possibilité de s'inscrire, de se connecter (s'il est déjà inscrit), d'ajouter un thème et de consulter un ou des thèmes. Le système quant à lui devait gérer l'inscription et afficher les thèmes aux utilisateurs.



Suite à l'avancement du projet, nous avons rajouter des fonctionnalités supplémentaires telles que la consultation des articles, l'ajout et le vote d'articles.

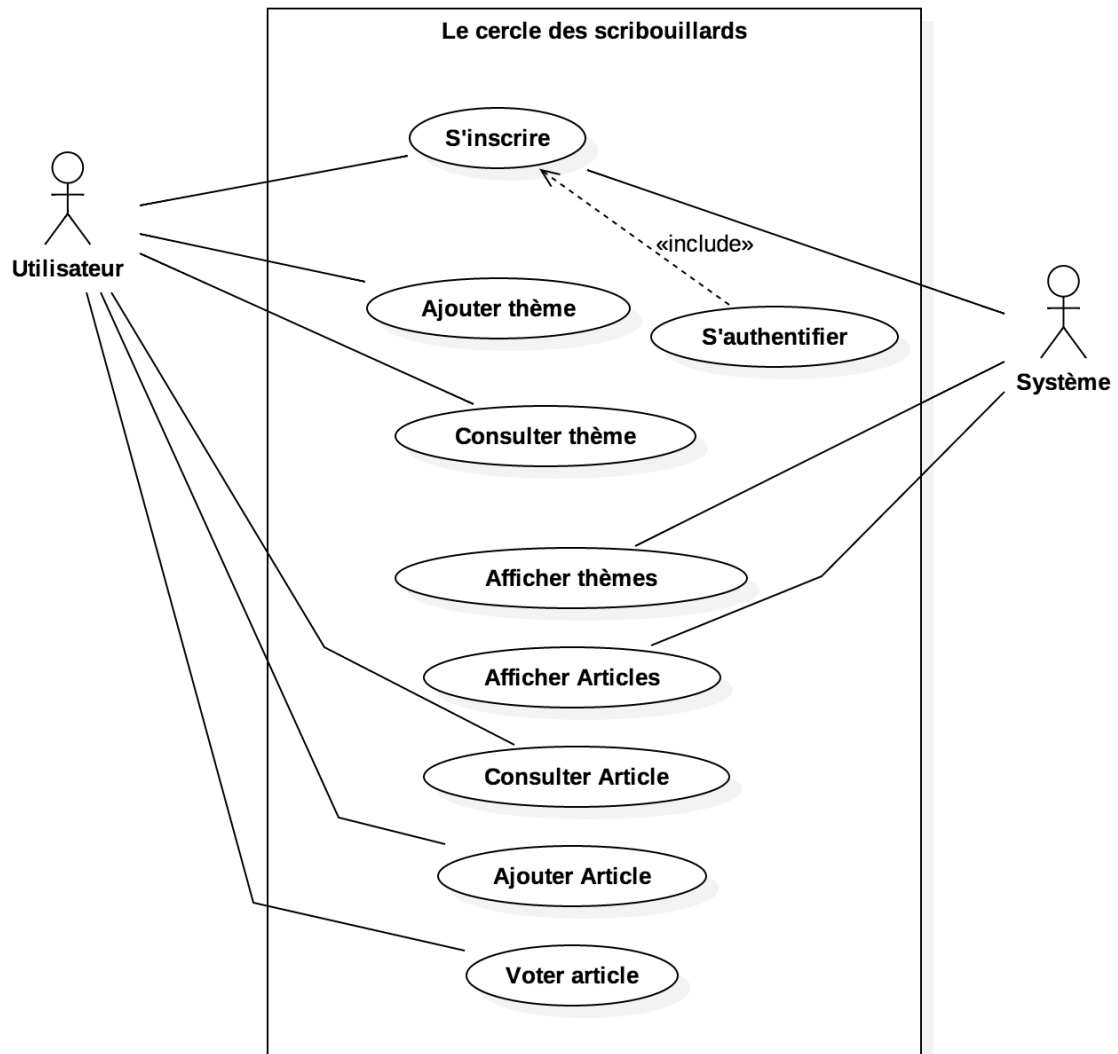
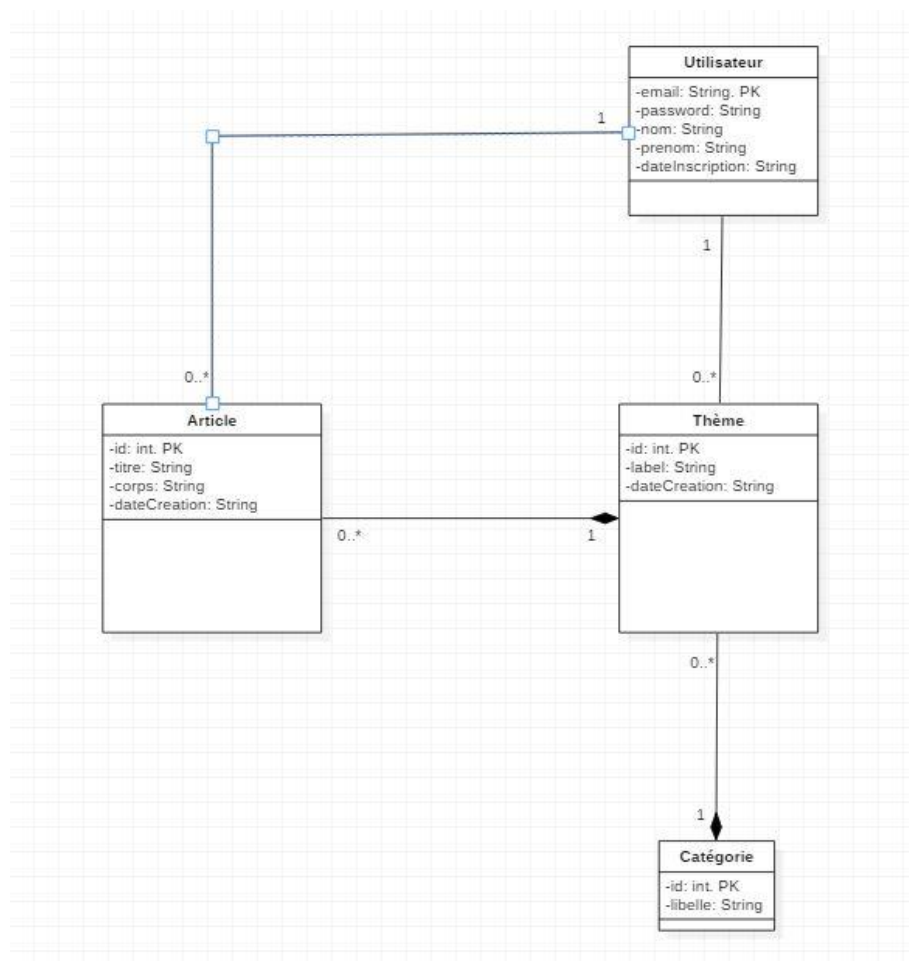
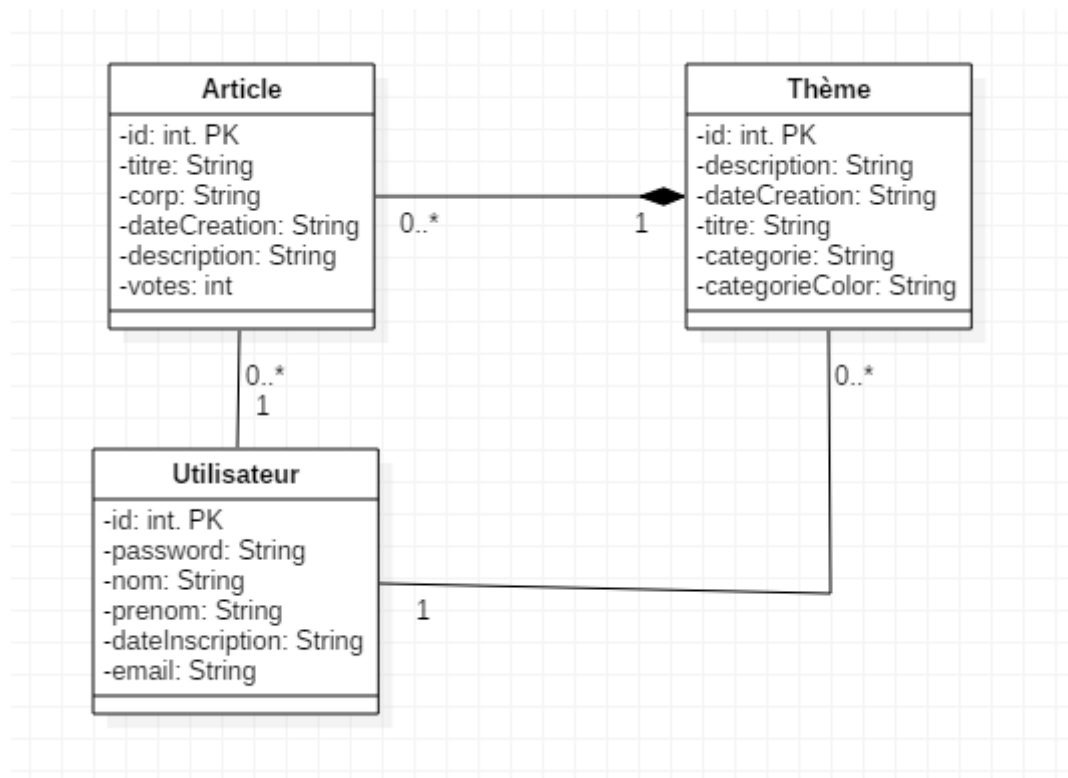


Diagramme de classes

Le second type de diagramme qui nous a été très utile est le diagramme de classe. En effet, ce diagramme est très utilisé en génie logiciel pour présenter les classes et interfaces des systèmes ainsi que les différentes relations entre celles-ci. Notre premier diagramme de classe est le suivant :



Puis par la suite, notre diagramme de classe a évolué et est devenu le suivant :



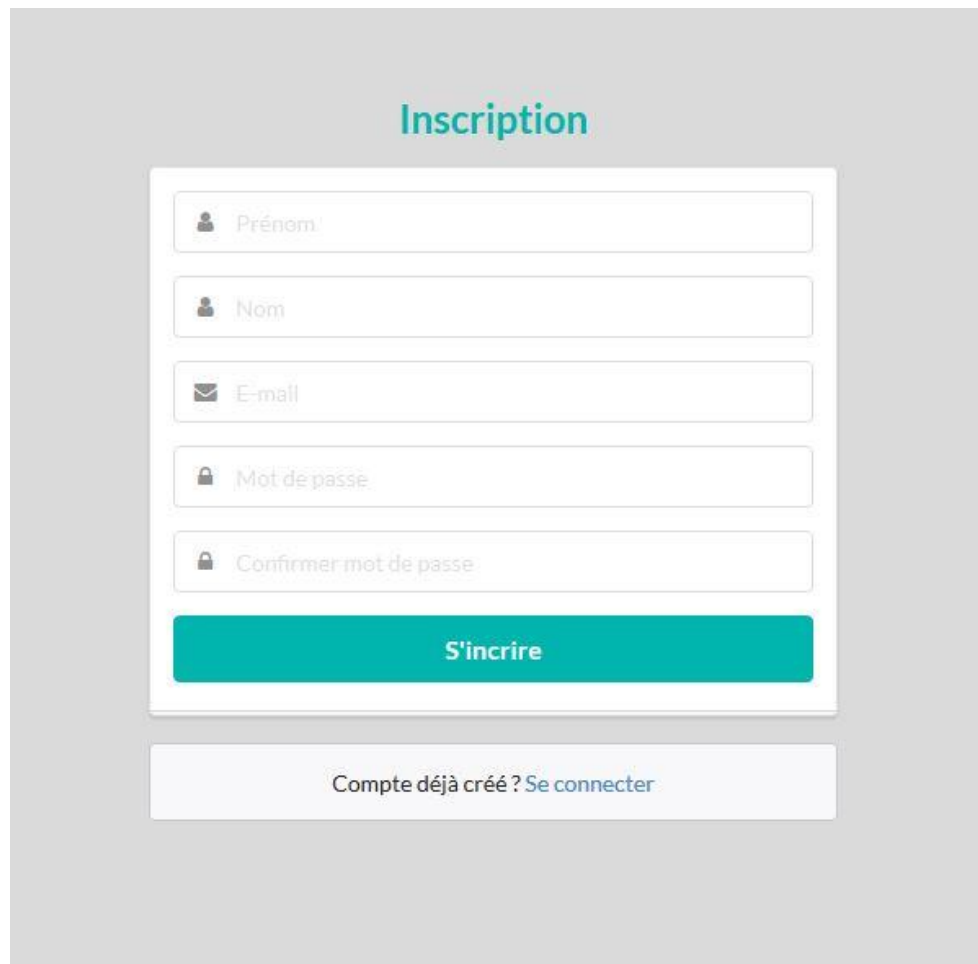
Nous avons d'abord fait le choix de stocker les catégories en base. Cependant comme les catégories sont statiques. Il s'agit d'une liste d'option fixe, nous avons trouvé plus judicieux de passer par une classe Java et de représenter les catégories sous forme de constantes de type String.

Ainsi la classe catégorie n'apparaît plus car il ne s'agit pas d'une entité. Elle est modélisée par une classe et elle est stockée en tant qu'attribut de la classe Thème.

Fonctionnalités

- L'inscription :

Les utilisateurs peuvent s'inscrire en remplissant le formulaire d'inscription ci-dessous :



The image shows a registration form titled "Inscription" in teal. The form is centered on a light gray background. It contains five input fields, each with a small icon on the left: a person icon for "Prénom", another person icon for "Nom", an envelope icon for "E-mail", a padlock icon for "Mot de passe", and another padlock icon for "Confirmer mot de passe". Below these fields is a large teal button labeled "S'inscrire". At the bottom of the form, there is a link that says "Compte déjà créé ? [Se connecter](#)".

La page ci-dessus se compose de 5 champs qui sont obligatoires à la saisie. Tout en bas on peut cliquer sur « Se connecter » pour accéder à la page de connexion si on dispose déjà d'un compte.

Lorsque les champs ne sont pas remplis le message d'erreur suivant s'affiche :

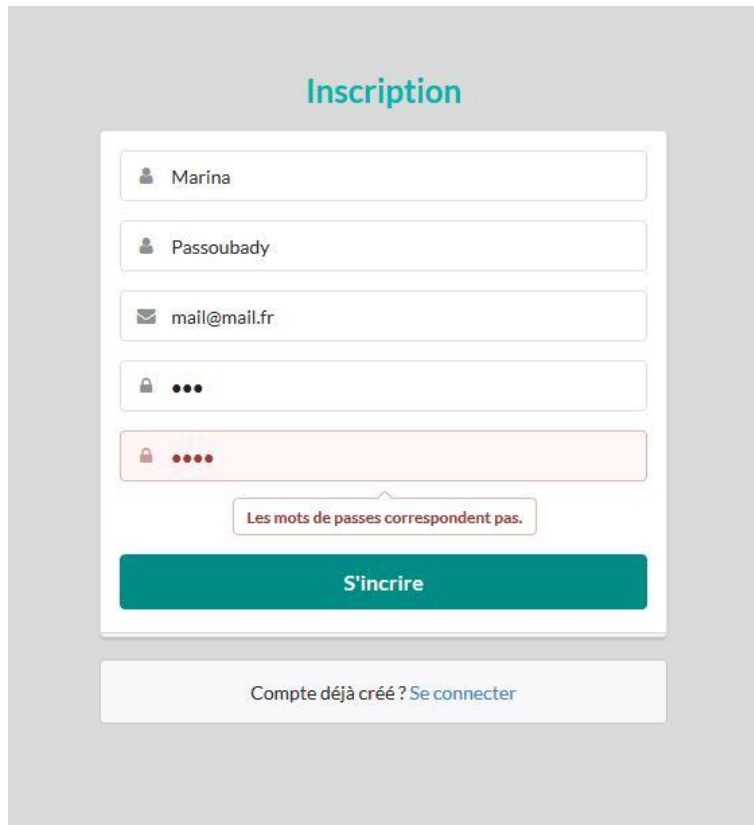


Lorsque le champ email est erroné on affiche le message suivant :



A screenshot of a form validation error. At the top, there is a light red rectangular box containing an envelope icon and the text "dfgh". Below this box, a small white callout bubble with a red border contains the text "Votre email n'est pas valide." in red.

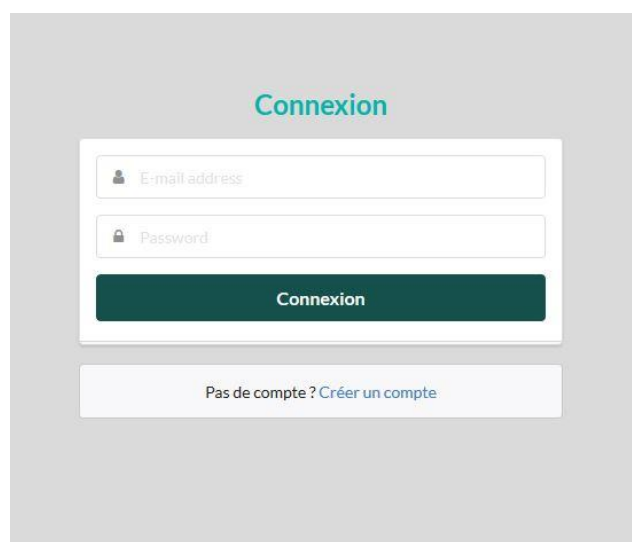
Quand les mots de passe ne correspondent pas lors de l'inscription on affiche le message :



A screenshot of a registration form titled "Inscription" in teal. The form contains five input fields: a name field with "Marina", a username field with "Passoubady", an email field with "mail@mail.fr", a password field with three dots, and a confirmation password field with four dots. The confirmation password field is highlighted in light red. Below it, a white callout bubble with a red border contains the text "Les mots de passes correspondent pas." in red. At the bottom of the form is a teal button labeled "S'inscrire". Below the form is a light gray box containing the text "Compte déjà créé ? [Se connecter](#)".

- La connexion :

Les scribouillards peuvent se connecter à partir de la page de connexion. Pour ce faire ils doivent remplir les champs login et mot de passe définis lors de leur inscription :



A screenshot of a login form titled "Connexion" in teal. The form contains two input fields: "Email address" and "Password". Below these fields is a dark teal button labeled "Connexion". At the bottom of the form is a light gray box containing the text "Pas de compte ? [Créer un compte](#)".

En cas d'oubli du login ou mot de passe, il faut cliquer sur « Créer un compte ».

Dès lors où les champs ne sont pas renseignés les messages suivants apparaissent :

Connexion

E-mail address

Ce champ est obligatoire.

Password

Ce champ est obligatoire.

Connexion

Pas de compte ? [Créer un compte](#)

Si l'email ou le mot de passe sont erronés on affiche ceci :

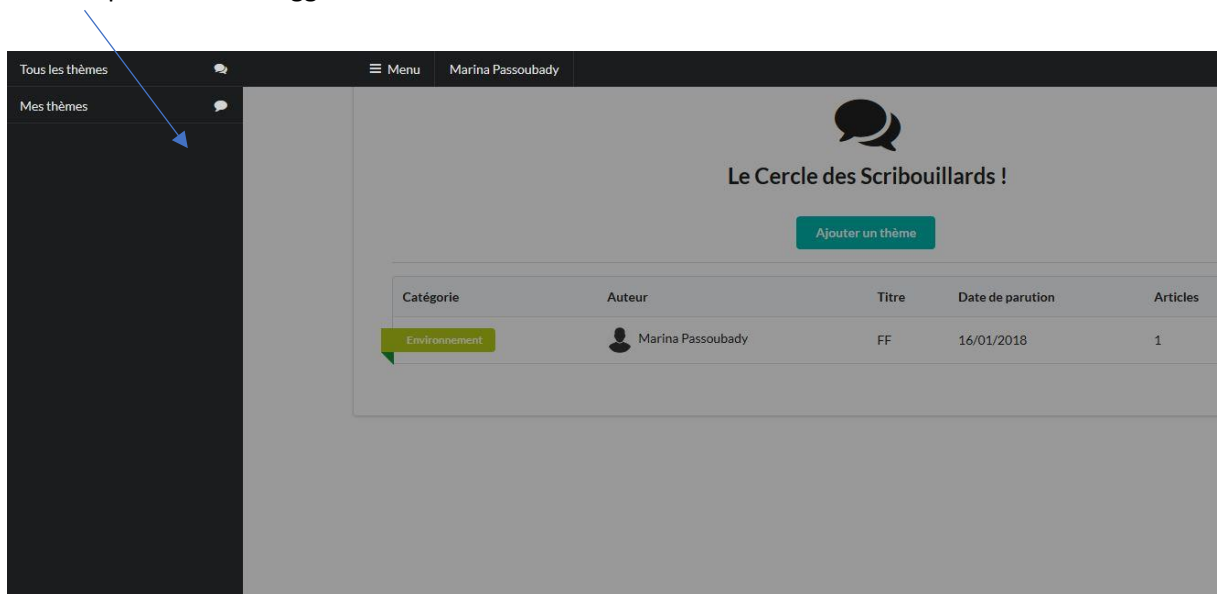
Erreur de connexion
Email ou mot de passe invalide

- La page d'accueil se présente de la manière suivante :



On peut voir plusieurs éléments dans la page d'accueil : la catégorie du thème, son auteur, son titre, sa date de parution, son nombre d'article et un œil pour pouvoir consulter les articles.

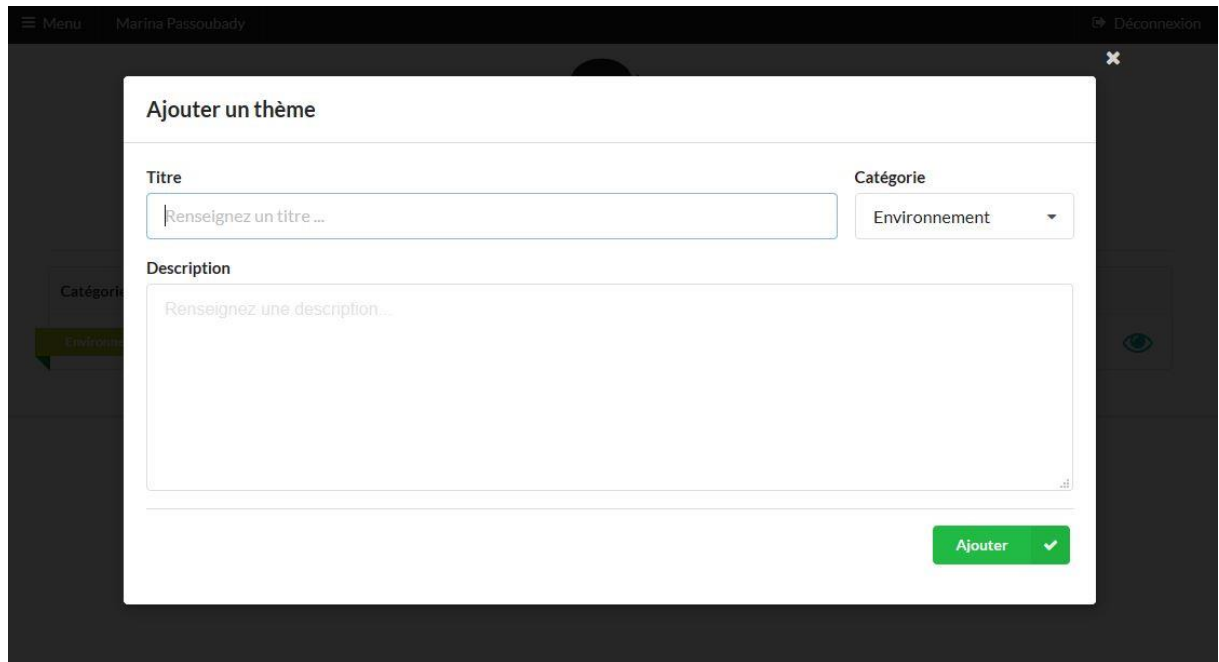
Elle est composée d'une Toggle sidebar :



Depuis le toggle sidebar on peut voir tous les thèmes disponibles dans le site en cliquant sur « Tous les thèmes ». En cliquant sur « Mes thèmes », on a la possibilité de voir tous les thèmes publiés par l'utilisateur connecté.

- La proposition de thème :

En cliquant sur le bouton proposer un thème, une fenêtre modale apparaît et permet aux scribouillards d'entrer leur thème ainsi que la description associée :



The screenshot shows a web application interface with a dark sidebar and a light main area. A modal window titled "Ajouter un thème" is open. It contains a "Titre" field with placeholder text "Renseignez un titre ...", a "Catégorie" dropdown menu currently set to "Environnement", and a "Description" text area with placeholder text "Renseignez une description ...". A green "Ajouter" button with a checkmark is at the bottom right of the modal.

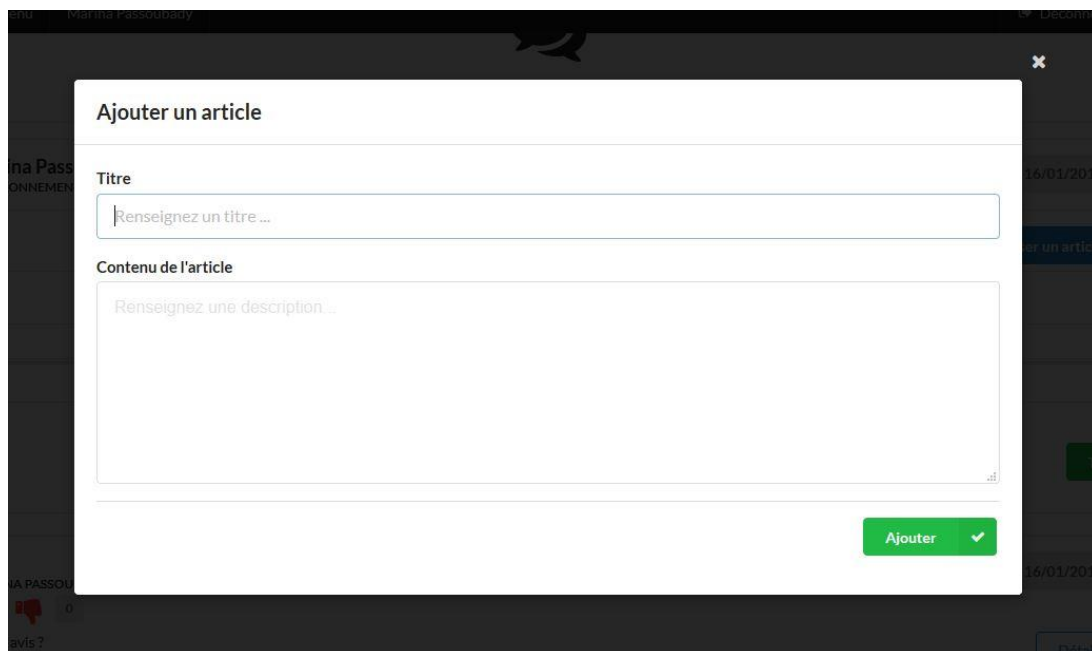
Si la proposition de thème échoué (champ non renseigné) le message suivant apparaît et l'action doit être réitérer par le scribouillard :



Erreur d'ajout. Veuillez renseigner tous les champs.

- La proposition d'articles :

Après avoir sélectionner un thème, un utilisateur peut proposer un article qui dispose d'un titre et d'un contenu :



The screenshot shows the same web application interface. A modal window titled "Ajouter un article" is open. It contains a "Titre" field with placeholder text "Renseignez un titre ...", a "Contenu de l'article" text area with placeholder text "Renseignez une description ...", and a green "Ajouter" button with a checkmark at the bottom right.

Si les champs ne sont pas renseignés et le formulaire est validé le message suivant apparaît :

Erreur d'ajout. Veuillez renseigner tous les champs.

- Lecture d'articles :

Notre application permet aux utilisateurs de lire les articles d'autres scribouillards en cliquant sur le bouton « œil » qui à l'aide d'un accordéon déroule le contenu de l'article :

PROPOSITIONS

Plus récents Plus votés Trier

Article 2
MARINA PASSOUBADY
16/01/2018
Lire l'article
Votre avis ?
Paragraphe 1 Paragraphe 2 bla bla contenu de mon article

Article 1
MARINA PASSOUBADY
16/01/2018
Lire l'article
Votre avis ?

- Le tri des articles par date :

On peut trier les articles par date de parution plus récente à moins récente et réciproquement :

PROPOSITIONS

Plus récents Plus votés Trier

Article 1
MARINA PASSOUBADY
16/01/2018
Lire l'article
Votre avis ?

Article 2
MARINA PASSOUBADY
16/01/2018
Lire l'article
Votre avis ?

- Les votes :

Les scribouillards peuvent évaluer un article à l'aide des « boutons-icône » j'aime et je n'aime pas. Nous avons mis à disposition les fonctions suivantes sur les votes :

- Tri de l'article le plus voté au moins voté

PROPOSITIONS

Plus récents Plus votés Trier

Article 1
MARINA PASSOUBADY
16/01/2018
Vous avez déjà voté !
Lire l'article

Article 2
MARINA PASSOUBADY
16/01/2018
Vous avez déjà voté !
Lire l'article

- Tri de l'article le moins voté au plus voté

PROPOSITIONS

Plus récents Plus votés Trier

Article 2
MARINA PASSOUBADY
16/01/2018
Vous avez déjà voté !
Lire l'article

Article 1
MARINA PASSOUBADY
16/01/2018
Vous avez déjà voté !
Lire l'article

- Fonctionnalité bonus :

Lors du Sprint 4, il nous a été demandé de rajouter un test de « Turing » à l'inscription, afin d'éviter les inscriptions frauduleuses à l'aide de robot.

Nous avons fait appel au service reCaptcha de Google. Nous l'avons intégré côté serveur afin d'éviter que ça soit désactivé si c'était côté client.

Tests unitaires

Nous avons réussi à couvrir 56% du code de notre application. Nous avons eu des difficultés à utiliser correctement les annotations Mockito. Grâce à Travis, à chaque modification du code, si le « build » ne passait pas c'était un signe que certains tests ne passaient plus.

De cette manière-là nous avons pu travailler sur les tests pour assurer qu'ils passent tous. Sonar indiquait le pourcentage de couverture pour chaque classe. Grâce à cet outil, nous avions pour objectif d'atteindre 100% de couverture pour les classes testées.

Nous avons eu cependant des difficultés à tester les repositories de manière unitaire.

Nous avons donc tenté de réaliser un test d'intégration des repositories. Depuis la classe de test, nous lançons l'application Spring et la création de la base de données. Comme nous utilisons un fichier «.properties », nous avons tenté de l'ajouter dans les ressources des tests mais nous n'avons pas réussi à l'intégrer au tests.





On a abandonné ces tests et nous nous sommes concentrés sur les autres classes. Au fur et à mesure nous avons effectué les tests. Ce qui nous a permis d'y revenir pour mieux les écrire et les modifier si le code avait changé.

Tests fonctionnels

Nous avons effectué des tests fonctionnels qui sont disponibles dans le cahier des charges. Nous les avons effectués manuellement. Il y a eu d'abord une approche sur l'outil Cucumber mais en vue de nos priorités, nous avons privilégié les autres types de tests et l'avancement du projet.

Difficultés rencontrées, solutions trouvées

Au cours de ce projet, nous avons pu rencontrer quelques difficultés telles que celles citées ci-après, mais nous avons su trouver les solutions nécessaires à leur résolution :

	Organisation 	Spring boot 	Test unitaire 	Sonar Cloud 
Difficulté	Recherche d'une méthode de travail adaptée.	Utilisation globale : La plus grande difficulté étant de se documenter correctement et suffisamment sur le sujet, les solutions étant immenses.	Résolution des dépendances des classes à tester.	Mise en place de l'intégration continue : nous avons tenté SonarCloud pour éviter l'installation de l'outil SonarQube à chacun. Problème au niveau du script, sur la commande de lancement de l'analyse. Après Plusieurs modifications des configurations, en vain. Un utilisateur avec des droits suffisants était nécessaires. Nous n'avons pas réussi à trouver la solution dans la documentation.
Méthode et Solution	Division de l'équipe en sous-groupe pour le développement et la documentation. Recherche de tutoriel sur l'outil Trello et entraide. Mise en place d'un tableau de bord Trello.	Recherches et tri sur les résultats. Exemple : Pour Spring Security, Il y a plusieurs façons d'instaurer une sécurité et il a fallu plusieurs semaines pour sélectionner la meilleure méthode.	Utilisation de la documentation Springboot sur les tests afin de savoir quelles annotations utiliser et comment les utiliser. Utilisation des annotations « Mockito », « when », « then » etc pour simuler des scénarios et que les tests puissent fonctionner.	Le professeur nous a attribué un compte sur son serveur Sonar. Nous avons modifié le script à l'aide du professeur en utilisant Travis pour lancer l'analyse à chaque commit du projet sur Git.

Points perfectibles, restes à faire, points non solutionnés

Par manque de temps, nous n'avons pas g rer les votes   l'aide de requ tes Ajax en mode asynchrone. Il aurait  t  int ressant de coder en javascript aussi pour ce projet et de toucher   la technologie Ajax.

Pour encore plus valoriser l'aspect communication et  change, outre les votes, qui restent anonyme, proposer un service de commentaire pour chaque article. Cela permettrait comme un forum d'interagir et d'exprimer son opinion.

L'ergonomie du site reste assez simple car nous ne voulions pas rendre le nom du site clich  sur le design. Cependant, certains choix de couleurs ou de placement aurait pu  tre am liorer pour favoriser l'exp rience du client sur notre site.

Un service de notification aurait  t  int ressant. Quasiment tous les sites proposent une notification par mail. Dans notre cas, un mail pourrait  tre envoy  pour confirmer son inscription sur le site mais  galement si un article a atteint plus de X likes.

Retours personnels sur le module

Retour globale de l'équipe des Scrib's

Nous trouvons que les cours du module sont très bien organisés et efficace. En effet, les cours montrent directement les pistes auxquelles nous devrions nous intéresser. L'enseignant est très concis et pragmatique ce qui est hautement apprécié par l'ensemble de l'équipe.

Par ailleurs, le fait que les cours soient centralisés sur un site dédié est génial. Cela nous évite de perdre du temps à les chercher sur le commun, et nous y donne accès partout.

Ce module nous a apporté une vision fraîche et à jour de ce qu'est le Génie Logiciel aujourd'hui ce qui est très important dans le cadre de la veille technologique de chacun. Enfin, Vincent Boutour est vraiment disponible pour ses étudiants et est de grande aide lors de difficultés rencontrées ou de blocages ce qui est appréciable.

Retour personnel de Sami BOUHAFS

Les points que j'ai aimés :

- Apprendre à utiliser des nouveaux outils (des outils très utiles)
- Travailler dans un groupe dans lequel j'ai pu m'épanouir (ne pas avoir de contrôle et se concentrer seulement sur le projet est une bonne chose)
- Faire des présentations orales qui rapportent des points (bonne source de motivation)
- La totale liberté que vous nous offrez (la confiance d'un professeur est très importante pour une bonne cohésion dans une classe)
- Vos retours personnels (les petites anecdotes) ainsi que vos nombreux conseils sont très appréciables car nous allons bientôt nous insérer dans la vie professionnelle

Les points que j'ai moins appréciés :

- Parfois les cours étaient un peu trop long (je conviens qu'il est primordial de passer par là)
- J'aurais aimé coder notre site en PHP car je maîtrise mieux que le Java
- J'aurais aimé que vous nous fassiez des petites prises en mains des nouveaux outils (des petites démos de github, trello ...)

Dans l'ensemble, j'ai énormément apprécié ce module car c'est vraiment le module qui parle à tous les élèves (MOA et MOE) et c'est l'un des modules où j'ai réellement appris de nombreuses choses (github, trello, spring ...).

Retour personnel de Bettina Aza

Lors de ce module j'ai pu découvrir et utiliser les outils GitHub, Trello et le langage markdown qui m'ont beaucoup servi lors de mon rôle dans le projet. En effet, j'étais chargée avec Sami de rédiger la documentation et notre planning. Grâce à ce projet j'ai également découvert l'utilité springboot, j'ai pu voir que ce Framework permettait une facilité de configuration et de développement de l'application. Je n'hésiterais pas à réutiliser dès que l'occasion se présentera tous les langages et outils découverts lors de ce module. Par ailleurs, j'ai bien aimé travailler avec mon groupe, j'ai trouvé que notre organisation en deux équipes a donné une fluidité au projet et nous a permis de mieux

nous connaître les uns les autres. En fin j'ai trouvé que la disponibilité des supports de cours sur un site internet était une très bonne idée.

Retour personnel de Marina Passoubady

J'ai beaucoup apprécié ce module. Dans un premier les cours sont très vivants avec un support clair et concis. Un professeur passionné qui arrive encore à jongler entre la passerelle vie étudiante et vie professionnel. On se sent très vite à l'aise et lui aussi compréhensible et proche de ce qu'on souhaite comme réponse. J'ai appris des choses mais surtout compris. Beaucoup de notion abordée dans le cours, j'en avais déjà entendu parler mais je n'avais pas d'opinion clair dessus. Ainsi M. Boutour nous a partagé son savoir sans nous l'imposer. Cette manière d'enseigner incite plus la curiosité de l'élève dans le bon sens.

Concernant le projet, nous étions 4 mais confiant. L'organisation et l'attributs des tâches et rôles de chacun s'est très vite mise en route. Fonctionnant par binôme pendant les séances et par Skype durant la semaine : niveau communication je l'ai trouvé fluide et efficace.

Nous avons choisi, en suivant les conseils du professeur, de partir sur un projet simple et ensuite de l'alimenter avec d'autres fonctionnalités. Cette vision a finalement porté ses fruits car nous sommes arrivés quasiment à nos objectifs. Chacun a contribué au projet, nous avons fait l'effort de travailler sur des technologies que nous n'avons pas expérimentées (Git, Markdown, Semantic UI, SpringBoot). J'ai beaucoup apprécié découvrir en même temps que de travailler sur le projet. J'ai découvert la puissance du framework Spring.

Au sein de l'équipe, une très bonne entente et du sérieux de la part de chacun. Pour gagner en production, nous avons travaillé sur des tâches où nous étions le plus à l'aise. Nous prenions ensuite le temps d'expliquer aux autres nos avancements. Git nous a été très utile pour ce projet, j'ai longtemps évité de l'utiliser durant le DUT. Dorénavant je m'exigerai de l'utiliser, je me suis rendu compte que cela facilite le partage du code lorsqu'on travaille à plusieurs régulièrement.

De même que Sonar pour l'intégration continue, un outil puissant qui nous a aidé pour trouver les bugs et couvrir nos tests. Une interface simple et flexible que j'utiliserai très souvent aussi.

Retour personnel de Mohamed LEBIB

- **Structures des cours** : C'est parfait. Les cours donnent une liste exhaustive des étapes à suivre pour réussir son projet. J'ai vraiment appris énormément de choses, allant des concepts de base en passant par les best practices et en finissant par les bons outils. Les cours apportent une vision concise, pragmatique et très à jour de ce qu'est le génie logiciel.
- **Projet** : Le nec plus ultra. Moi qui veux faire du Backend Java en Spring Boot / Hibernate, j'ai vraiment été bien servi. J'ai aussi découvert comment faire de l'intégration continue, qui est plus qu'essentiel pour garantir la qualité logicielle. L'équipe a été géniale. Nous nous sommes répartis les tâches de façon très spontanée. La communication a été optimale (aucune friture dans l'oreillette). Tout cela orchestré par M. BOUTOUR, qui a été très présent et très à l'écoute pour nous.
- **J'aurais aimé** : Qu'on aille un peu plus en profondeur avec Scrum. La présentation de cette méthode agile a été un peu brève dans le cours. Vraiment dommage !
- **Pour résumer** : La liste des cours que j'ai appréciés durant mon parcours (Prépa intégrée, L2 puis L3) n'est pas très longue. Celui-ci fait désormais partie de cette liste. Merci beaucoup Vincent.

Conclusion

Nous avons trouvé ce projet très enrichissant. En effet, il nous a permis de découvrir ou d'approfondir de nombreux points. Nous avons pu développer des compétences organisationnelles : Le fait d'avoir divisé le groupe en deux sous équipes nous a permis d'accélérer nos recherches et d'établir une meilleure communication. N'ayant jamais travaillé ensemble tous les 4, la division a obligé chacun à partager et prendre connaissance de l'avancement du travail des autres. Cette organisation a bien fonctionné dans le groupe.

Nous avons essayé de travailler en agile en fonctionnant par sprint. Nous avons vite pris l'habitude de vérifier nos « to do list », d'établir des séances de brainstorming et de redéfinir les objectifs de la semaine.

Par ailleurs nous avons pu découvrir l'utilisation de nouveaux outils utiles lors de projets de groupe tels que GitHub, Spring et Trello. Nous en sommes convaincus, cela nous sera très utile lors de nos prochains projets à plusieurs. Ce projet nous a imposé de suivre des bonnes pratiques qu'on négligeait ou ignorait auparavant. Cela nous a donné goût à voir au-dessus d'une simple application qui « fonctionne ».

Nous sommes désormais persuadés que les contraintes du projet sont des atouts et des outils nécessaires à sa bonne réalisation. Ce module a été d'un très bon équilibre entre consignes et réalisation en groupe.