

MarkUs, une application web d'annotation du code des étudiants

Benjamin Vialle, Ghislain Guiot

École Centrale de Nantes

RMLL - 10/07/2013



Sommaire

- 1 Introduction
- 2 Contexte
- 3 Impact sur l'enseignement et l'apprentissage
- 4 Déploiement de MarkUs
- 5 Conclusion

Des besoins identifiés

Motivation

Comment **gérer** et **évaluer efficacement** les travaux rendus par les étudiants en TP/Projet ?

Usage de MarkUs

- Déployé à l'École Centrale de Nantes depuis septembre 2010
- L'École Centrale de Nantes contribue au développement depuis l'été 2009
- Terrains d'utilisation
 - Enseignements d'informatique (rapport et code)
 - Promotions de plus de 350 étudiants
 - Plus de 20 enseignants impactés

Sommaire

- 1 Introduction
- 2 Contexte**
- 3 Impact sur l'enseignement et l'apprentissage
- 4 Déploiement de MarkUs
- 5 Conclusion

Limites des dispositifs traditionnels

Du côté des enseignants

- Gros **volume** de soumissions à traiter (plusieurs centaines par TP)
- Difficulté d'**harmonisation** des facteurs de correction d'un chargé de TD/TP à l'autre
- Gestion papier
 - Amoncellement de piles
 - Retour des dossiers aux étudiants
- Gestion par courriels
 - Erreurs dans le destinataire
 - Archives .zip illisibles
 - Lourdeurs

Limites des dispositifs traditionnels

Du côté des étudiants

- Difficulté pour **recupérer/consulter** ses travaux corrigés
- Gestion papier
 - Perte de rapports
 - **Partage** de la copie avec son binôme ?
- Gestion par courriels
 - Erreurs dans le destinataire
 - Un courriel parmi d'autres

MarkUs, un outil de correction en ligne de travaux étudiant

MarkUs? Mark us!

MarkUs est :

- Application **Web**
- Destiné à l'évaluation de projet informatique
- Dépôt **versionné** des travaux des étudiants
- **Annotation directe** des documents par les enseignants
- Diminution du **temps** de correction

Organisation autour de MarkUs

L'équipe de MarkUs

Karen Reid, enseignante à l'Université de Toronto, responsable de l'équipe
Morgan Magnin, enseignant chercheur à l'École Centrale de Nantes,
encadre les projets d'étudiants français

- 4 développeurs principaux
- Équipe trimestrielle d'étudiants (Canadiens et Français)
- Utilisation de MarkUs par plusieurs Universités (Canadiennes et Française)
- Développement collaboratif sur GitHub
- Projet dirigé par les demandes des clients et les projets étudiants

Quelques fonctionnalités

Amélioration de l'enseignement (correcteur)

Possibilité d'annoter

- Code source (avec coloration syntaxique)
- Images
- PDF

The screenshot displays the MarkUs interface for grading a C program. On the left, a code editor shows the source code for a function named 'lancerTex.c'. The code includes comments and a loop structure. A yellow highlight is placed over a line of code, and a text box annotation reads 'Mettez des noms de variables explicites'. The right side of the interface features a 'Notes' panel with a table of feedback rules. The table has columns for 'Noms', 'Régularité', and 'Règle d'erreur'. The feedback rules are categorized into 'Analyse des résultats et affichage', 'Fonctions', and 'Algorithmes'. Each rule includes a grade (e.g., 2.0 Possible, 0.0 Très insuffisant) and a description of the error or requirement.

Noms	Régularité	Règle d'erreur
Analyse des résultats et affichage		
2.0 Possible	Les résultats sont mal expliqués	
0.0 Très insuffisant	Les résultats ne sont pas donnés	
1.0 Insuffisant	Les résultats sont donnés sans explications	
2.0 Possible	Les résultats sont mal expliqués	
3.0 Bonne	Les résultats sont expliqués avec des implications	
4.0 Excellent	Les résultats sont bien expliqués	
Fonctions		
0.0 Non	Définition des fonctions, avec quelques erreurs	
0.0 Très insuffisant	Peu de fonctions	
1.0 Insuffisant	Utilisation des fonctions fournies	
2.0 Possible	Spécifications des fonctions correctes	
3.0 Bonne	Définition des fonctions, avec quelques erreurs	
4.0 Excellent	Fonctions bien implémentées	
Algorithmes		
0.0 Non	Difficilement compréhensible ou beaucoup d'erreurs	
0.0 Très insuffisant	Peu d'algorithmes ou incompréhensible	
1.0 Insuffisant	Difficilement compréhensible ou beaucoup d'erreurs	
2.0 Possible	A peu près compréhensible avec quelques erreurs	
3.0 Bonne	Peu d'erreurs ou quelques algorithmes	
4.0 Excellent	Algorithmes très bien écrits	
Lisibilité (à titre indicatif, indésirable, commentaires, nom des variables)		
0.0 Non	Peu lisible - beaucoup d'éléments pas clairs	
0.0 Très insuffisant	Étrange	

Figure : Vue du correcteur

Quelques fonctionnalités

Amélioration de l'enseignement (correcteur)

- Critères fixes d'évaluation
- Annotations (code source, images et pdf)
- Plusieurs correcteurs pour une copie
- Gestion des correcteurs par critères



Figure : Détermination de critères

Quelques fonctionnalités

Amélioration de l'enseignement (correcteur)

- Prise en charge de plusieurs TP, dans l'idée d'une instance de MarkUs par matière
- Gestion des **échéances** avec pénalités de retard (configurables)
- Possibilité de voir et corriger une **ancienne** version

Quelques fonctionnalités

Amélioration de l'enseignement (élève)

- Constitution des groupes en fonction des TP
- Export des commentaires
- Retour amélioré et plus rapide
- Possibilité de revoir les commentaires
- Possibilité d'une *remark request*

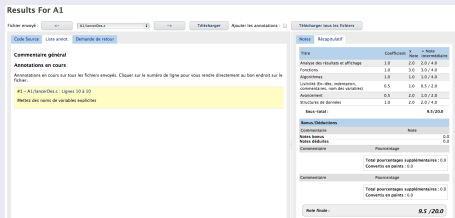


Figure : Vue des résultats par les étudiants

Quelques nouvelles fonctionnalités

Release de MarkUs 1.0

- Compatibilité avec Ruby 1.9.3 et Ruby on Rails 3.x
- **Gestion des sections** au sein d'une promotion
- Conversion des PDF **instantanée**
- Ajout des *remark requests*
- **Nouveau tableau de bord** pour l'administrateur

Démo

Et si nous passions à une petite illustration pratique...

Sommaire

- 1 Introduction
- 2 Contexte
- 3 Impact sur l'enseignement et l'apprentissage**
- 4 Déploiement de MarkUs
- 5 Conclusion

Pourquoi MarkUs séduit les enseignants

- Gestion de **gros volumes**
- Gestion **centralisée** des documents
- **Diminution du temps** de correction (environ 50%)
- **Dématérialisation**
- Accès **nomade**

Pourquoi MarkUs séduit les étudiants

- Une **unique** plate-forme de soumission et de correction
- Accès **permanent** aux anciens travaux annotés par les enseignants
- Amélioration du **délai** d'obtention de la correction

Du côté de Centrale Nantes

Déploiement du logiciel pour les cours d'informatique

- Depuis septembre 2010
- Interconnecté avec **LDAP**
- Utilisé en 1ère et 2e année :
 - 370 et 340 étudiants impactés
 - 21 enseignants concernés
- Enseignements d'**informatique** :
 - Algorithmique
 - C
 - Java

Les effets bénéfiques de MarkUs

Côté étudiants :

- Effet pédagogique du **respect des dates limites**
- Chaque **individu** accède à la correction du travail de son groupe
- **Consultation accrue** des corrections laissées par les enseignants

Les effets bénéfiques de MarkUs

Côté enseignants :

- Meilleure gestion **logistique**
- Une première **uniformisation** des critères de correction
- Aspect **incitatif** de la correction

Sommaire

- 1 Introduction
- 2 Contexte
- 3 Impact sur l'enseignement et l'apprentissage
- 4 Déploiement de MarkUs**
- 5 Conclusion

Autour de MarkUs

Modalités pratiques

- Écrit en Ruby, avec Ruby on Rails
- Documents sauvegardés via Subversion
- Accès via l'application web
- Utilisateurs avancés : accès CLI via une API REST

Essayez le !

- Machine virtuelle : instance de MarkUs pré-configurée

Sommaire

- 1 Introduction
- 2 Contexte
- 3 Impact sur l'enseignement et l'apprentissage
- 4 Déploiement de MarkUs
- 5 Conclusion**

Synthèse

MarkUs, une application web d'annotation du code des étudiants

Comment améliorer la procédure d'évaluation des TP/projets d'étudiants ?

Usage de MarkUs

- Logiciel **libre**
- Annotation du **code**, des **.pdf** et des **images**
- Facilité de prise en main
- Seul coût : installation et maintenance
- Utilisation plébiscitée par les étudiants et les enseignants

Améliorations à venir

Vers un élargissement de l'utilisation de MarkUs

- Module d'**annotation tactile**
- Intégration d'annotations mathématiques
- Tests automatiques du code envoyé par les étudiants
- Élargissement à d'autres matières
- Intégration à un **ENT** ?
- Déploiement facilité à l'aide de machines virtuelles

Plus d'informations

Liens et contacts

- Site du projet : <http://markusproject.org>
- Essayer le logiciel en ligne : <http://demo.markusproject.org>
- Sources : <https://github.com/MarkUsProject/Markus>
- Blog EAT-TICE de l'Ecole Centrale de Nantes :
<http://eat-tice.ec-nantes.fr>
- Chan IRC : #markus sur irc.freenode.net
- Mailing list : markus-dev@markusproject.org