



TPE: Travail Personnel Encadré

Domaine : CSCL / ACAO – Apprentissage collaboratif assisté par ordinateur

Introduction au Codage collaboratif avec Codeboard

Jean Baptiste FLORIAL

Mars 2017

Encadrant

[NGUYEN Trong Khanh] | [HO Tuong Vinh]



Un IDE basé sur le Web pour enseigner la programmation en classe. Créez et partagez facilement des exercices avec les élèves. Analyser et inspecter les soumissions des élèves en un seul clic.

Guide rapide d'utilisation

Codeboard.io

[Explore](#)

[Docs](#)

[Sign in](#)

[Sign up](#)



[No account? Sign up here.](#)

Sign in

A web-based IDE to teach programming in the classroom.

Easily create and share exercises with students.

Analyze and inspect students' submissions with a single click.



Sommaire -----2

Chapitre I:

- Introduction -----3
- Scenario -----3

Chapitre II - Vue d'ensemble de Codeboard:

- À propos de Codeboard -----4
- Les modules de Codeboard -----4
- Configuration system requise -----5
- Technologie -----5
- Installation & Configuration (Pour les utilisateurs avancés) -----6
- Langages de programmation supportée -----8
- PRÉSENTATION GÉNÉRALE -----9

Chapitre III - Codeboard en Pratique

- Création d'un projet sur Codeboard -----12
- Configuration d'un projet sur Codeboard -----13
- Coder sur Codeboard -----16
- Gestion des soumissions -----17
- Evaluation -----21

Conclusion : -----21

References -----21



CHAPITRE 1 : Intro.

[1] INTRODUCTION

L'apprentissage collaboratif consiste en une variété de pratiques éducatives dans lesquelles les interactions entre pairs sont caractérisées par le partage et la collaboration. Cette méthode peut-être appliquée dans le cadre des cours de programmation, un environnement où les étudiants travaillent ensemble sur des projets de programmation, faire des travaux collaboratifs dans un environnement d'apprentissage collaboratif pour pouvoir apporter des améliorations dans l'élaboration des exercices pratiques en classe et à la maison.

L'objectif est d'améliorer l'enseignement de la programmation en adoptant un modèle d'apprentissage collaboratif, avec l'aide des outils éducatifs qui donnent de l'animation et de la visualisation du programme. Et aussi la collaboration entre les étudiants pendant le développement du programme au cours de programmation.

[1.1] SCENARIO APPLICATIF :

Le codage collaboratif joue un rôle important dans la formation des programmeurs, du point de vue enseignant ou apprenant. Un outil puissant mais léger comme Codeboard aide à la réalisation des travaux et leçons pratiques et attrayantes sur la programmation et respecte les scénarios d'apprentissage ci-dessous.

1- Les MOOC ou les FAD jouent un rôle important dans la façon dont les étudiants d'aujourd'hui apprennent la programmation. Les plates-formes de formation existantes fournissent des services pour l'hébergement de contenus de conférences, de vidéos et de quiz simples. Cependant, ils ne soutiennent pas les enseignants dans la création d'exercices de programmation sophistiqués et gradués automatiquement. Codeboard compte à combler cette lacune, permettant aux enseignants de fournir des exercices où les étudiants n'ont pas besoin de télécharger des fichiers, n'ont pas besoin d'installer un logiciel, mais obtenir des commentaires immédiats sur la correction de leur travail. Codeboard peut être utilisée dans plusieurs plateformes de formation, avec succès en l'étendant à des cours avec plus de 60.000 étudiants.

2- Distribuer facilement un exercice de programmation (en partageant son URL) permet un style d'enseignement différent dans les contextes traditionnels: après l'introduction d'un nouveau concept aux étudiants, les enseignants peuvent fournir un court exercice de 2-3 minutes qui récapitule le nouveau concept. En utilisant les fonctionnalités d'un tel outil pour accéder aux projets ou présentations des élèves, les enseignants peuvent facilement mettre en place la solution d'un étudiant sur un projecteur sans devoir passer d'ordinateurs portables ou échanger des fichiers.

3- Travailler les présentations des exercices de programmation des étudiants par courrier électronique ou comme imprimés sur papier peut être fastidieux et de travail intensif pour les enseignants. Les présentations intégrées d'un tel outil rendra cette tâche beaucoup plus facile. Les soumissions sont automatiquement classées et facilement accessibles pour les enseignants car elles peuvent être ouvertes et exécutées directement dans l'IDE de. Cela peut économiser beaucoup de temps lors de la vérification des dizaines d'exercices chaque semaine.



4- En outre, les processus de correction des erreurs et La démonstration pour chaque élève n'est pas optimale. Beaucoup Les élèves répètent souvent les mêmes erreurs. Par conséquent, l'instruction pour chacun prend beaucoup de temps. En particulier le professeur est presque impossible de suivre les progrès de la pratique De chaque élève de donner des conseils appropriés ou de Conférences en conséquence.

Objectif de ce Guide utilisation.

Tenant comptes de tous les scenarios ci-dessous nous allons montrer (Les Fonctionnalités de Codeboard) comment l'outil Codeboard peut les réaliser.

CHAPITRE 2: Vue d'ensemble de Codeboard.

Cette section fournit un bref aperçu de L'Environnement de développement collaboratif intégrer Codeboard.

[2.1]À propos de Codeboard:

Codeboard est un projet de recherche développé par des ingénieurs logicielles de l'ETH Zurich Christian Estler et Martin Nordio.

Un IDE basé sur le Web conçu et tester sous un système d'exploitation Linux. Un outil facile à utiliser pour créer enregistrement, et partagez facilement des exercices de programmation.

Codeboard IDE est un environnement de codage collaboratif qui fournit les caractéristiques suivantes:

- Partager les codes
- Travailler sur les mêmes codes en même temps
- classer automatiquement une soumission.
- classement automatique à l'aide des tests unitaires.
- classement automatique à l'aide d'une chaîne de résultat.
- Cloner un projet

[2.2] Les modules de Codeboard :

- Calibrage automatique des soumissions basées sur des tests unitaires ou sur des pilotes de test personnalisés.
- Ouvrez et inspectez toute soumission directement dans l'IDE Codeboard.
- Suivez les progrès des élèves en utilisant des statistiques spécifiques au projet.
- Créez des exercices dans un IDE riche en fonctionnalités et partagez-les simplement en utilisant leurs URL.
- Masquer tout fichier ou dossier que les étudiants ne devraient pas voir.
- Utilisez votre langue d'enseignement préférée: C, C ++, Eiffel, Haskell, Java, Python.
- Outils Externe (LTI): permet d'intégrer des outils supplémentaires grâce à la technologie LTI.



[2.3] Configuration system require:

Etant une application Web, les simples utilisateurs de Codeboard n'ont besoin que d'un Navigateur Web pour accéder au site web : <https://codeboard.io>

The image shows the Codeboard.io website header with the logo and navigation links: Explore, Docs, Sign in, and Sign up. Below the header is a 'Sign up' form. The form has a title 'Sign up' and four input fields: 'Username*', 'Email*', 'Password*', and 'Confirm password*'. Each field is a simple text box. Below the fields is a blue button labeled 'Create account'. At the bottom of the form, there is a link: 'Already have an account? [Sign in here.](#)'

Figure 1 : Interface de création de compte sur Codeboard

The image shows the Codeboard.io website header with the logo and navigation links: Explore, Docs, Sign in, and Sign up. Below the header is a 'Sign in' form. The form has a title 'Sign in' and two input fields: 'Username or Email' and 'Password'. Each field is a simple text box. Below the fields is a blue button labeled 'Sign in'. At the bottom of the form, there is a link: 'Forgot your password? [Reset your password here.](#)'

Figure 2 : Interface d'authentification de Codeboard



[2.4] Technologie :

Codeboard est un outil Open Source : Libre, Gratuite distribué sous la licence MIT, (code disponible pour être modifier).

Exigences : Codeboard requiert

- NodeJS,
- MySQL, MongoDB et graphicsmagick (pour redimensionner des images de profil utilisateur).
- Nodejs : testé avec la version 0.12.9
- MongoDB : testé avec la version 2.6.4
- Codeboard a été testé sur un système Ubuntu 14.04.

[2.5] Installation & Configuration (Pour les utilisateurs avancés)

Pour ceux qui souhaitent utiliser Codeboard sur leur propre server locale ou internet peuvent télécharger les codes sources de l'application (<https://github.com/codeboardio/codeboard>) effectuez le déploiement et apporter vos propre modifications ou amélioration

Préparation du serveur, Nous devons installer MySQL et Mongo DB pour créer une base de données

- 1- Créer les deux bases de donnée avec le nom Codeboard avec les droits suffisants : MySQL data base est utilisé pour les informations système (IDE projet) Mongo DB data base est utilisé pour authentification.
- 2- Source MongoDB: <https://www.digitalocean.com/community/tutorials/how-to-install-mongodb-on-ubuntu-14-04>
Source MySQL : <https://www.garron.me/en/bits/create-database-user-set-grant-privileges-mysql-console-terminal.html>
- 3- Cloner le référentiel sur votre Machine serveur

```
git Clone https://github.com/codeboardio/codeboard.git
```

- 4- Changez dans le dossier Codeboard et installez toutes les dépendances

```
cd codeboard
```

```
# Installer toutes les dépendances du serveur
```

```
npm install
```

```
# Assurez-vous d'avoir Bower installé : ça permet d'installe les versions correctes des packages dont nous avons besoin et leurs dépendances.
```

```
Sudo npm install -g bower
```

```
# Installer toutes les dépendances client
```

```
bower install
```

```
Codeboard utilise Grunt pour automatiser diverses tâches. Assurez-vous d'avoir installé le Grunt-CLI
```

```
Sudo npm install -g grunt-cli
```



Configuration de Codeboard

- 5- Codeboard nécessite un certain nombre de paramètres, tels que les noms de bases de données, user login mots de passe, etc. Toutes ces configurations doivent être définies dans les fichiers suivants

Lib / config / env / all.js
Lib / config / env / development.js
Lib / config / env / production.js
Lib / config / env / test.js
Exécuter et tester Codeboard

Utilisez la commande suivante pour exécuter Codeboard (en mode de développement) :

Important

```
grunt serve
```

Construire une version d'optimisation pour le déploiement de production

```
# Crée un dossier dist  
# Déployer à partir de dist en utilisant la commande: NODE_ENV = noeud de production  
server.js  
grunt build
```

Test Codeboard : pas nécessaire

```
# Exécuter des tests côté client
```

```
Grunt test: client
```

```
# Exécuter des tests côté serveur
```

```
Grunt test: serveur
```

NB : Codeboard est très sensible à la gestion de version, et ne prends pas en comptes les versions ultérieures.



[2.6] Langues de programmation supportée :

La plateforme prend en charge différents langages de programmation et différents types de projets. Tous les compilateurs, programmes et outils sont exécutés sur un système Ubuntu Linux de 64 bits.

Langage	Version	Détails
C	4.8.4	GCC
C++	4.8.4	G++
C++14	6.2.0	G++
Eiffel	15.11	ISE EiffelStudio version
Eiffel-SCOOP	15.08	ISE EiffelStudio using SCOOP for concurrency
Haskell	7.10.3	Glasgow Haskell Compiler
Haskell-HSpec	7.10.3	Glasgow Haskell Compiler
Infer-Java	0.4.0	Facebook Infer for Java, with Java 1.7.0
Java	1.8.0	Oracle Java SE
Java-JUnit	1.8.0	Oracle Java SE, JUnit 4.11
Python	2.7.6	NumPy and SciPy packages pre-loaded
Python-UnitTest	2.7.6	NumPy and SciPy packages pre-loaded
Reason	0.0.1	More at http://facebook.github.io/reason



[2.6] PRÉSENTATION GÉNÉRALE

Espace principale

L'espace de travail de Codeboard est similaire à celui d'autres programmes d'environnement de développement intégré à base web. Il est divisé en différentes barres de menus. Dans cette espace vous pouvez créer un nouveau projet, consulter les anciens projets et voir les informations concernant, consulter les soumissions, modifier les paramètres du profil, ouvrir un projet dans l'IDE, voir les statistiques, et modifier les paramètres du projet.

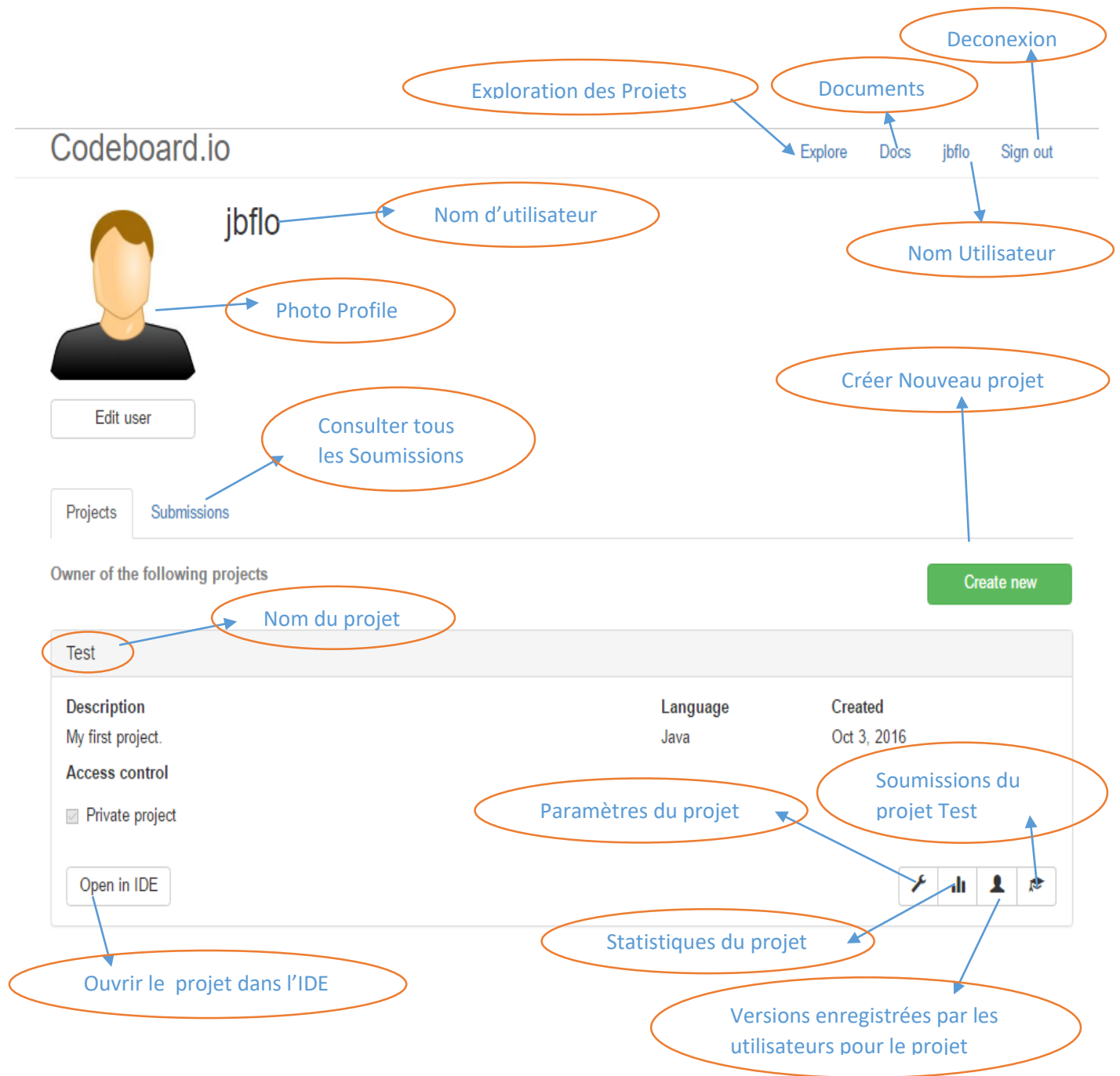


Figure 3 : Espace (utilisateur) principale de CB



Explore : Ici vous pouvez avoir accès à tous les projets créés avec accès public, essayé de relever des défis et de résoudre un des projets challenge.

Docs : Ici vous avez accès à la documentation de Codeboard, en anglais vous pouvez voir comment paramétrer vos projets, gérer les Soumissions des projets, configurer les Contrôles d'accès (public vs. privé), voir les Statistiques du projet, et...

Edit User : Ici vous pouvez modifier l'ensemble des paramètres utilisateur Photo profile, nom utilisateur, Modification du mot de passe.

Settings : Ici on a accès à la configuration générale du projet, gérer les paramètres d'accès

A screenshot of a web form for editing a user profile. At the top, there are two tabs: 'Public profile' (selected) and 'Password'. The form contains a 'Profile picture' section with a placeholder image of a person and a button 'Upload new picture' above a dashed box with the text 'Or drop a picture here'. Below this are several text input fields: 'Email (publicly displayed on your profile)', 'Full name', 'Web', 'Institution', and 'Located in'. At the bottom of the form is a blue button labeled 'Save changes'.

Figure 4 : Modifier les paramètres utilisateurs



Espace Secondaire (Espace de travail) (IDE) : cette espace fonctionne comme un IDE normale avec des fonctionnalités comme : ajouter supprimer modifier un fichier ou un dossier, Compiler exécuter et enregistrer ou soumettre le projet. Ici vous pouvez même partager le projet.

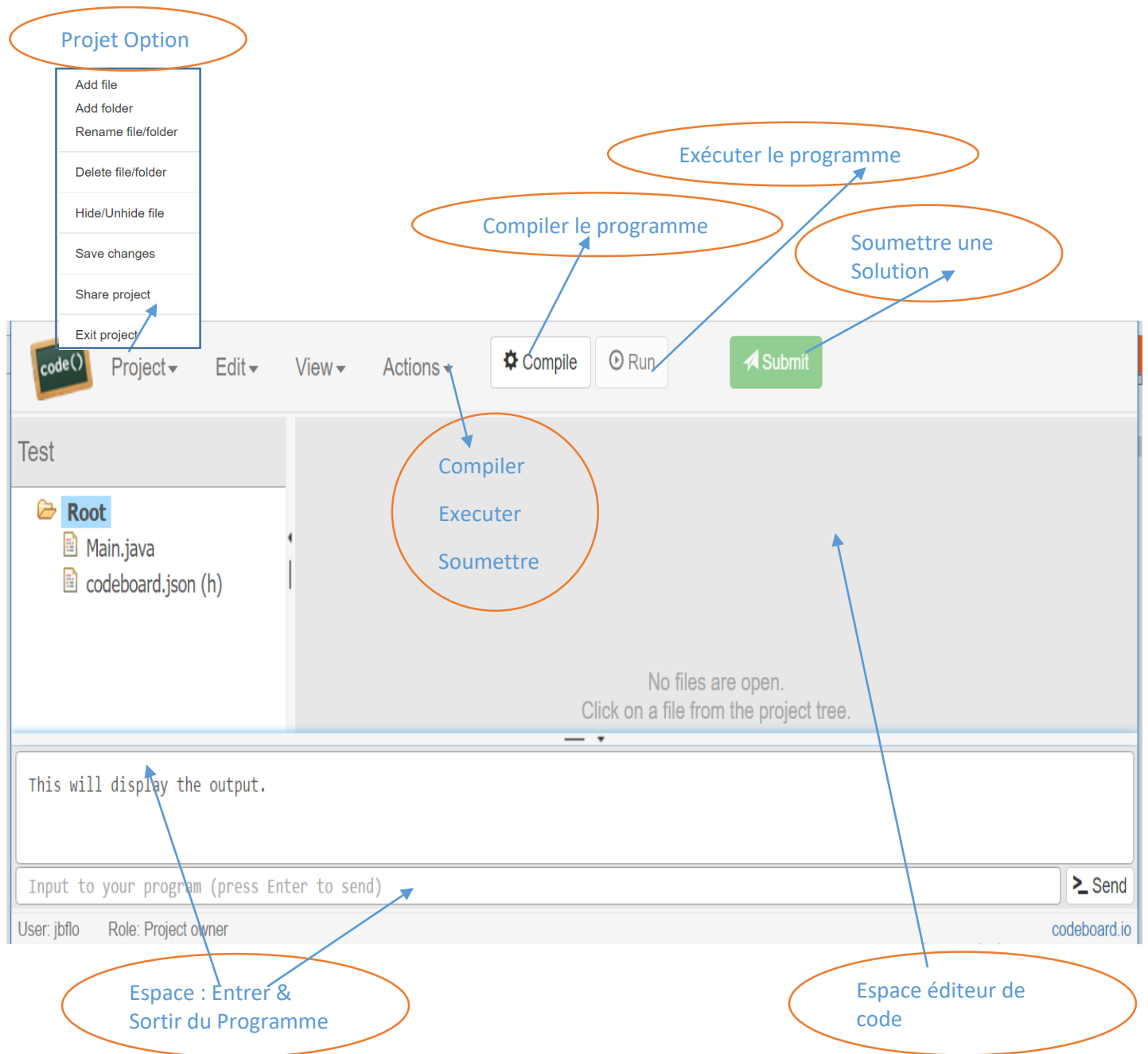


Figure 5 : Espace de codage



CHAPITRE 3 : Codeboard en Pratique

Cette section fournit des détails sur l'utilisation de l'environnement de développement collaboratif intégrer Codeboard, comment créer un projet et faire des travaux collaboratif.

[3.1] Création d'un projet sur Codeboard

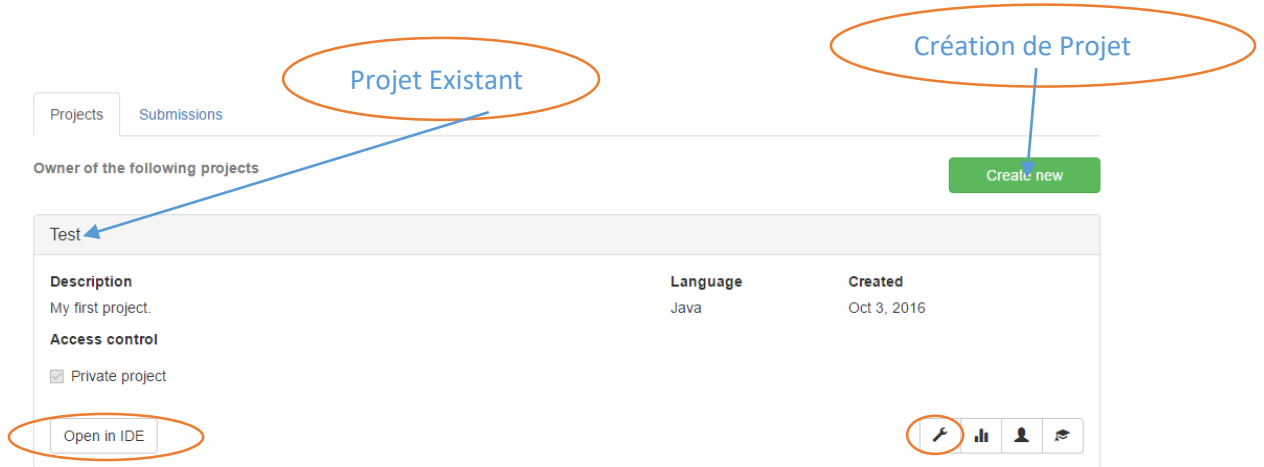


Figure 6 : Onglet Création d'un projet sur Coardboard

Lors de la création d'un nouveau projet sur Codeboard Vous pouvez maintenant définir les paramètres suivants : Nom et Description du projet Langage de programmation Accessibilité (projet public ou privé)

The screenshot shows the 'New project' form in Codeboard.io. The form has the following fields and controls:

- Project name***: A text input field.
- Description**: A text area with a placeholder.
- Language**: A dropdown menu currently showing 'Java-JUnit'.
- Access control**: A checkbox labeled 'Private project (only accessible by owners and users)'.
- Create**: A green button.
- Cancel**: A white button.

Figure 7 : Fenêtre création de projet

Vous pouvez créer un projet prive accessible aux utilisateurs que vous choisissiez, ou un projet public accessible à tout le monde.

Priver ou Publique on peut travailler de manière Collaborative sur un projet Codeboard



[3.2] Configuration d'un projet sur Codeboard

Une fois le projet créé nous pouvons configurer ou faire les modifications appropriées. Voir Figure 3 (Paramètre Projet)

Settings for **project 48706**

General Access Control Submission LTI Cloning Deletion

Project name*

CB

Description

Programming C++

Language

C++

Save changes

Figure 8 : Fenêtre paramétrage du projet créée

- 1- L'onglet général nous donne accès de modifier les informations clés comme types de langage utilisé etc...
- 2- L'onglet Accès Control est l'option qui nous permet d'ajouter et collaborer à d'autres utilisateur ou des anonymes

Settings for **project 48706**

General Access Control Submission LTI Cloning Deletion

☒ **Private project**
Only users listed below can access the project.

Project owners
Full control over the project. Access to submissions made by users. Access to personal versions saved by users.

jbflor ✕

Username of owner (separate multiple by comma)

Add

Figure 9 : Fenêtre Ajouter un utilisateur en tant que propriétaire

Noter qu'un projet Codeboard peut être privé ou public : en cas de projet Privé, pour effectuer des travaux collaboratif il faut ajouter manuellement les utilisateurs avec différentes droit d'accès.



Noter que les utilisateurs propriétaires du projet ont tous les droits qui existent.

Project users

Can open project in IDE. Can't see hidden files. Can't access project settings nor statistics. Changes to code are save as personal version.

Add

Figure 10 : Fenêtre Ajouter un simple utilisateur

Les simples utilisateurs ont des droits un peu limite : peuvent apporter et soumettre des modifications personnel, mais ne peuvent pas modifier le projet original. Ne peuvent pas modifier les paramètres du projet. Et peuvent pas voir les statistiques.

Project assistants

Like Project Owners but can't save changes to the project. Can't delete the project.

Add

Save changes

Activate

Figure 11 : Fenêtre Ajouter un utilisateur en tant qu'assistant

Et en fin les assistants, qui sont des utilisateurs très limités, ils apportent leur assistance aux propriétaires du projet sans avoir les droit de supprimer ou de modifier dans le projet.

- 3- Le paramètre de soumission donne aux propriétaires d'autoriser ou pas les soumissions des utilisateurs. Un exemple de scenario envisageable : la date limite de remise des TP sur le system est dépassé le propriétaire du projet (le professeur) peut interdire l'accès de soumission à un étudiant.

Codeboard.io

Settings for project 42866

General

Access Control

Submission

LTI

Cloning

Deletion

☒ **Allow users to submit their code**

Submissions allow for automatic grading and evaluating users' implementations.

Save changes

Figure 12 : Autoriser les soumissions



4- Pour la configuration des LTI (Learning Tools Interoperability). Outils d'apprentissage Interopérabilité permettant d'accéder à un projet Codeboard depuis une plateforme LMS, LTI permet aux plateformes éducatives de s'intégrer de manière sécurisée aux outils hébergés à l'extérieur. Codeboard a été testé compatible avec le Mooc EDX et le LMS Moodle. Les configurations sont assez simples.

- 1- Configuration depuis Codeboard : nous devons premièrement autoriser LTI pour le projet en question.
- 2- **LTI consumer key** et **LTI consumer secret** sont à remplir et sont fournis par l'administrateur de la plateforme LMS.

General Access Control Submission **LTI** Cloning Deletion

☒ **Allow LTI access**
Important: if LTI access is allowed, LTI-users can access the project regardless of access control settings.

LTI consumer key

LTI consumer secret

Url for LTI access
<https://codeboard.io/lti/projects/26655>

Save changes Discard changes

Figure 13 : LTI configuration

- 5- Cloning : on peut créer un nouveau projet avec la même structure des configurations du projet, cette action est appelée clonage puisque ça donne un projet typiquement identique.
- 6- Deletion : cet outil procède à la suppression de tous les fichiers et paramètres du projet, suppression de toutes les soumissions faites pour ce projet, suppression de toutes les versions sauvegardées par les utilisateurs pour ce projet.



[3.2] Coder sur Codeboard

Les projets créés sont listés dans l'espace utilisateur, l'onglet (Open in IDE, voir figure 6) permet de travailler sur un projet en traversant dans l'espace de travail (voir Figure 5).

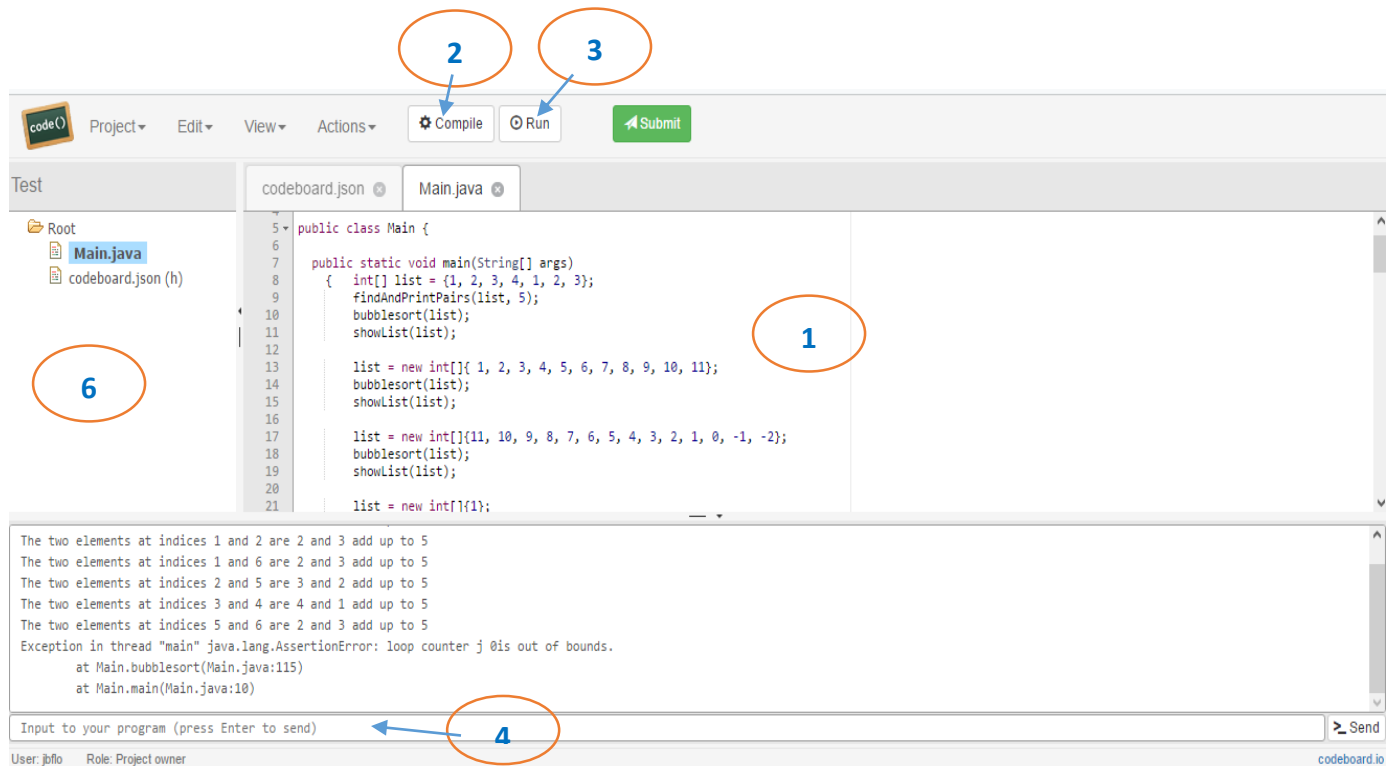


Figure 13 : LTI configuration

Un projet est créé initialement avec 2 fichiers à la racine :

- Le fichier Main.java : contient la fonction principale du programme
- Le fichier codeboard.json : fichier de configuration de projet

Partie 1 : Ecrire vos Code dans l'espace d'édition.

Partie 2 : Compiler le programme.

Partie 3 : Exécuter le programme

Partie 4 : Input (Entrer du programme)

Partie 5 : Output (Résultat de sorti du programme)

Partie 6 : Répertoire du Projet, créer nouveau fichier et dossier.



[3.2] Gestion des soumissions :

Codeboard.io

Explore Docs jbflo Sign out

jbflo

Onklet Soumission

Edit user

Projects Submissions

Liste des Soumission

Submission	Project	Created	Result	Tests passed	Tests failed	
101107	TestC	Feb 28, 2017 8:57:16 PM	-	-	-	Open in IDE
101109	TestC	Feb 28, 2017 8:57:34 PM	-	-	-	Open in IDE
101127	TestC	Feb 28, 2017 9:27:03 PM	-	-	-	Open in IDE

Figure 14: Gestion des soumissions

La Figure ci-dessus correspond à l'espace de travail utilisé par les propriétaires des projets pour contrôler les soumissions des projets publics, cette espace est utilisé par le enseignant pour gérer les soumissions multiple des étudiants. Open in IDE est l'onglet qui permet à l'enseignant de consulter chaque soumission des étudiants.

Les Types de Soumission :

Submissions for project 42866

Filter by username

#anonymous (1)

Created	Result	Tests passed	Tests failed	LTI	
Feb 28, 2017 1:16:25 PM	-	-	-	yes	Open in IDE

Figure 14.1 : Soumission par un utilisateur anonyme

Projet 42866 : l'identifiant du projet, ce numéro permet de trouver le projet en faisant des recherches

jbflo (3)

Created	Result	Tests passed	Tests failed	LTI	
Feb 28, 2017 9:27:03 PM	-	-	-	yes	Open in IDE
Feb 28, 2017 8:57:34 PM	-	-	-	no	Open in IDE
Feb 28, 2017 8:57:16 PM	-	-	-	no	Open in IDE

Figure 14.2 : Soumission par l'utilisateur identifier (jbflo)



testjb (2)				
Created	Result	Tests passed	Tests failed	LTI
Mar 2, 2017 4:25:10 PM	-	-	-	yes Open in IDE
Feb 26, 2017 11:51:33 PM	-	-	-	no Open in IDE

Figure 14.3 : Soumission par l'utilisateur testjb

[3.3] Statistiques :

Codeboard permet de voir les statiques sur l'ensemble du projet et est mesurer en fonction de la date de début ou de fin :

- Statistique sur Compilation et l'exécution du programme : le tableau ci-dessus présent le nombre de fois que le projet a été compiler et exécuter : 0 fois entre en Mai. Compiler 4 fois et exécuter une fois en juin.

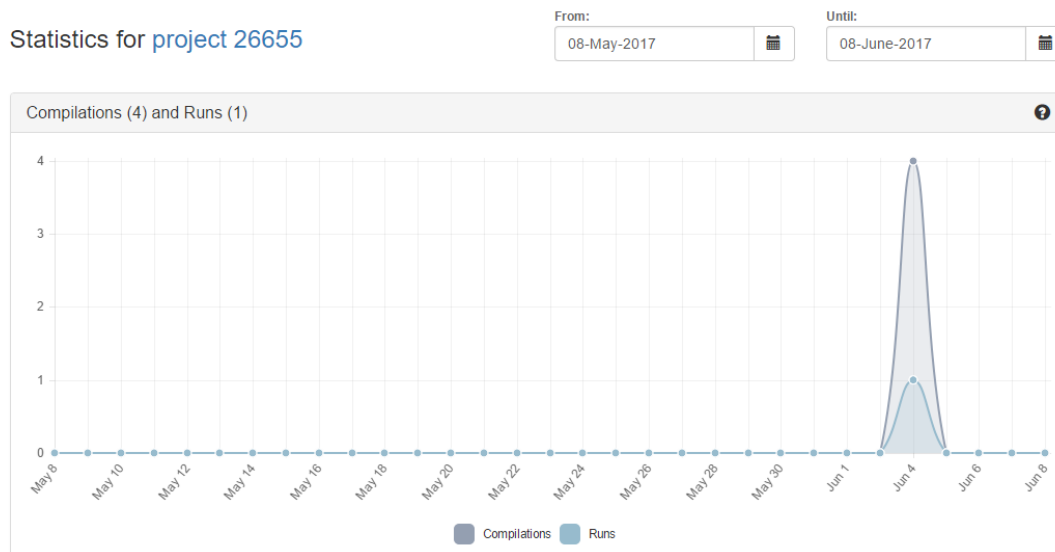


Figure 14 : Statistique sur Compilation et l'exécution

- Détails sur les utilisateurs : de manière plus détaillé codeboard donne l'information sur quel utilisateur qui a exécuter ou compiler le projet. très facile à adapter en salle de classe pour savoir quel étudiant a travaillé sur le projet.



Compilations (5)		
Username	Last compilation	# of compilations
jbflo	Jun 8, 2017 9:32:30 AM	5

First Previous 1 Next Last

Runs (1)		
Username	Last run	# of runs
jbflo	Jun 4, 2017 3:17:51 PM	1

First Previous 1 Next Last

Figure 15 : l'utilisateur Compilation et l'exécution

- Codeboard donne aussi des statiques sur le nombre de fois qu'un utilisateur a accédé au projet

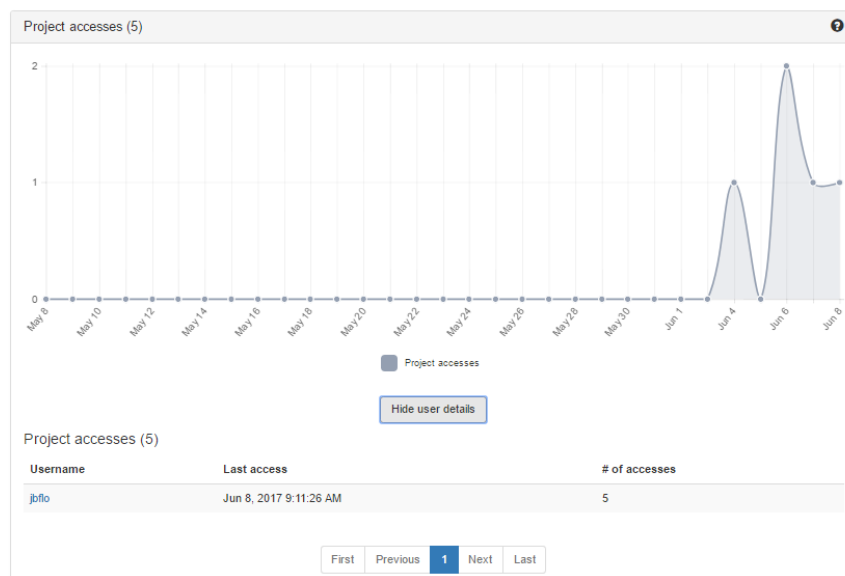


Figure 16 : Statistique sur l'accessibilité du projet

Codeboard donne aussi des statiques sur le nombre de fois qu'un utilisateur a fait une soumission

[3.4] Gestion des Version :

Codeboard donne la possibilité à chaque utilisateur d'enregistrer sont propres version du projet.et chaque version reste unique et accessible au propriétaire du projet principale.



[3.5] Evaluation :

Codeboard permet de créer des cas de test qui permet de valider une soumission. Dans le cas où l'outil est intégré dans un LMS comme Moodle, Codeboard peut envoyer les informations de soumission (Failed or Success) qui va permet au enseignant de faire l'évaluation.

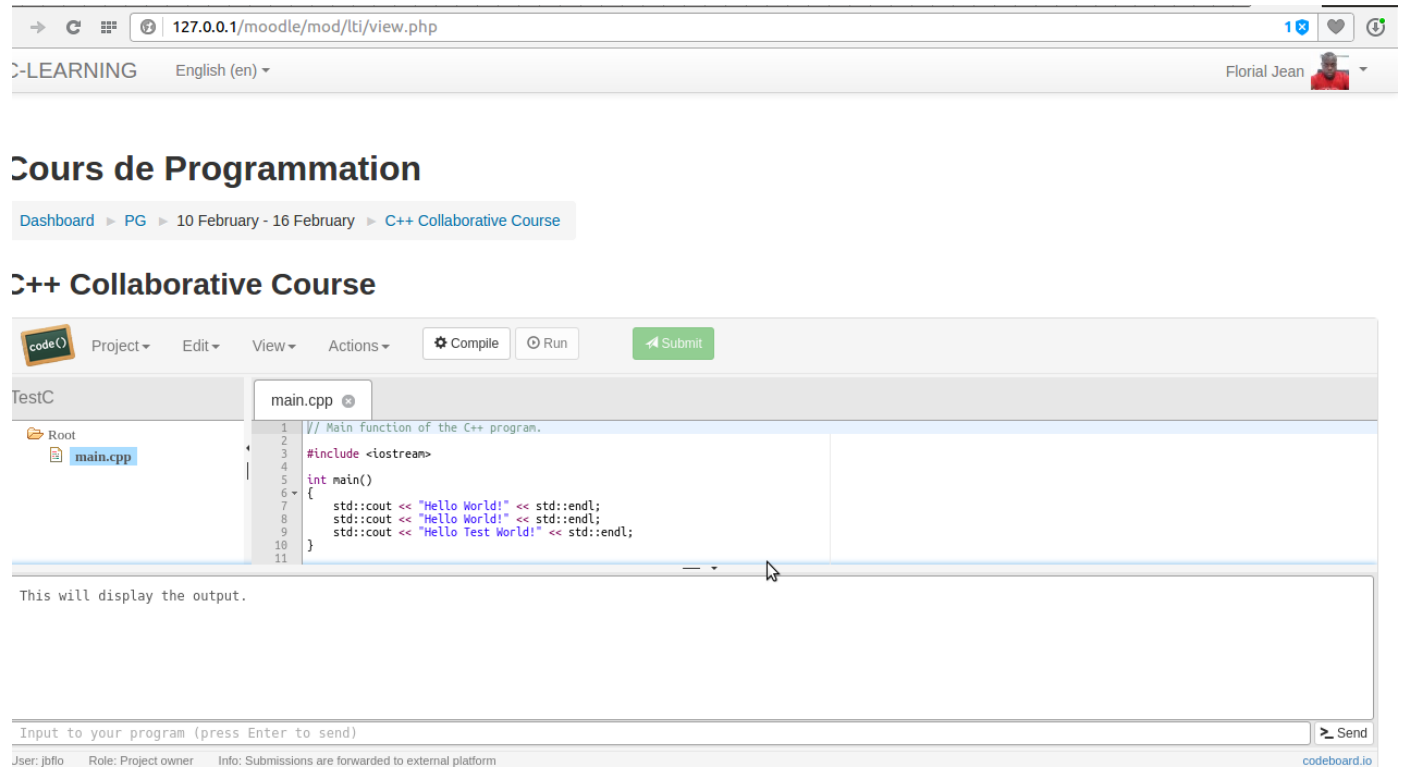


Figure 17: Intégration de Codeboard dans Moodle

[3.5.1] Evaluation sur les participations :

L'enseignant peut voir le nombre de fois qu'un étudiant a accédé au projet et le nombre de fois qu'il a compilé ou exécuté le programme. (Voir figure 15 & 16) de là l'enseignant peut donner une note de participation.

[3.5.1] Evaluation sur le travail effectué (Ligne de code Ajouter) :

Pour le moment Codeboard ne donne pas de statique sur le nombre de ligne de code ajouté par les utilisateurs du projet. Mais presente d'autres types d'évaluation .



[3.5.1] Evaluation sur le travail effectué (Soumission Avec compilation succès) :

Codeboard permet de faire l'évaluation sur l'ensemble de soumission effectué un classement automatique. Le formatage du classboard peut automatiquement classer une soumission. Actuellement, deux modes de classement sont pris en charge

- Classement automatique à l'aide d'une chaîne de résultat.
- Classement automatique en utilisant des tests unitaires.

Une note doit être une valeur entre 0 et 1. La valeur représente le pourcentage décrivant à quel point la soumission a été classée. Par exemple, une valeur de qualité valide est 0.42 (dont Codeboard peut également afficher 42%).

- Classement à l'aide d'une chaîne de résultats: chaque projet (et langage de programmation) disponible dans Codeboard peut implémenter le classement automatique en imprimant simplement une chaîne spéciale comme dernière sortie lorsque le projet est exécuté. Cette chaîne doit être sous la forme:

```
<!--@test=the_grade_value;num_tests_passed;num_tests_failed;-->
```

Lors de la soumission, Codeboard compile et exécute le programme soumis et vérifie si la sortie du programme se termine par une chaîne satisfaisant au format de la chaîne de résultats. Si une chaîne de résultats est trouvée, la valeur de la note (entre 0 et 1), le nombre de tests passés et le nombre de tests échoués est extrait de la chaîne de résultat et stocké dans les détails de la soumission.

- Classement à l'aide de tests unitaires: Codeboard prend en charge les tests unitaires pour un certain nombre de langages de programmation. Les types de projets avec support intégral pour les tests unitaires sont les suivants:

Java-JUnit

Haskell-HSpec

Python-UnitTest

L'utilisation des tests unitaires simplifie le classement automatique. Lors de la soumission, Codeboard exécutera tous les tests unitaires disponibles dans le "dossier de test de soumission". Le dossier de test de soumission est défini dans le fichier de configuration "codeboard.json", dans la propriété "DirectoryForTestSubmissionFiles". La note (valeur comprise entre 0 et 1) est calculée automatiquement en fonction du nombre de cas de test qui passent et échouent dans ce dossier.



Pour les affectations de programmation, il est recommandé de marquer tous les fichiers du dossier de test de soumission comme cachés. Si les élèves veulent écrire leurs propres cas de test, ils peuvent le faire en utilisant le dossier spécifié dans le fichier "codeboard.json" sous "DirectoryForTestFiles".

Variable d'environnement

Chaque fois que Codeboard exécute une soumission, il définit une variable d'environnement spéciale dans l'exécution-sandbox du projet.

Cette variable d'environnement et sa valeur sont:

```
CODEBOARD_ACTION_TYPE= submission
```

Limitation: la variable d'environnement n'est actuellement pas définie pour les projets avec des tests unitaires:

Java-JUnit

Haskell-HSpec

Python-UnitTest

En utilisant cette variable d'environnement, les propriétaires de projets peuvent implémenter des testeurs qui se comportent différemment si les utilisateurs déclenchent une action "soumettre" plutôt qu'une action "exécuter". Cela est particulièrement utile pour les projets qui ne supportent pas les tests unitaires.

[4] Conclusion :

Ce projet est disponible sous licence MIT. Qui nous permet d'utiliser la documentation et le code a notre guise. Codeboard est un outil très intéressant mais qui mérite d'être entretenue pour pouvoir être implémenté sur les systèmes d'exploitation mise à jour. C'est l'une des références en matière de formation collaboratif pratique. Avec ces outils nous avons une intégration presque parfaite d'utilisation des LTI, Codeboard intégrer dans Moodle dans le cadres d'un cours de programmation a été teste dans la version en ligne de l'outil, et les résultats obtenue réponds aux attentes des fonctionnalités cité. Ces résultats sont présentés dans le rapport final de ce travail.

Références

[1] <https://codeboard.io/docs/>

[2] <https://codeboard.io/projects/8450>