



Procédure de peer-pédagogie

Documentation

Staff pedago pedago@42.fr

Résumé: Ce document détaille la procédure de création d'un projet par les étudiants de 42 pour les étudiants de 42.

Table des matières

| | | |
|------------|--|-----------|
| I | Introduction | 2 |
| II | Définition d'un projet | 3 |
| II.1 | Le sujet | 3 |
| II.2 | Le barème | 5 |
| II.3 | La vidéo | 6 |
| III | Le processus de création d'un nouveau projet | 7 |
| IV | La plateforme de peer-pédagogie | 9 |
| IV.1 | Prérequis | 9 |
| IV.2 | Authentification | 10 |
| IV.3 | Utilisation de la plateforme | 10 |
| | IV.3.1 Le dépôt <code>projects</code> | 11 |
| | IV.3.2 Le submodule <code>templates</code> | 12 |
| | IV.3.3 Le submodule <code>scale_validator</code> | 12 |
| V | Conclusion | 14 |

Chapitre I

Introduction

L'école 42 existe aujourd'hui grâce à une notion forte : la peer-pédagogie. Le professeur du système scolaire classique est remplacé par une communauté étudiante au centre de son apprentissage. Il est donc naturel dans une structure comme la nôtre que les étudiants qui le désirent puissent soumettre des projets de leur invention à la communauté.

Que cette idée de projet provienne d'une connaissance déjà acquise par l'étudiant ou bien que l'étudiant choisisse de s'intéresser à un nouveau thème particulier dans le but de créer un nouveau projet n'importe pas. A la fin, la communauté a accès à un ensemble de projets de plus en plus vaste et de plus en plus riche.

Vous savez faire quelque chose qui n'est pas encore dans la liste des thèmes abordés ? Lancez-vous et partagez ! Vous voulez absolument découvrir une techno absente de la liste des thèmes abordés ? Lancez-vous et partagez !

Votre nombre et votre diversité sont votre force. Vous pouvez laisser une trace dans l'histoire de notre école !

Vous êtes 42 !

Chapitre II

Définition d'un projet

Le projet est une unité d'apprentissage de notre école, au même titre qu'une piscine ou un examen par exemple. Toutefois, les projets sont l'unité la plus importante car ils sont présents tout au long de votre scolarité ; les projets évoluent avec vous et vous avez un impact réel sur leurs contenus et leur diversité.

À 42, un projet est constitué de trois éléments :

- Le sujet
- Le barème
- La vidéo

II.1 Le sujet

Un sujet à 42 est un document pdf généré à partir d'une source écrite en [LaTeX](#). Ce format permet une mise en page puissante et élégante. Vous trouverez très facilement sur internet des milliers de tutoriels pour apprendre LaTeX. Vous êtes bien entendus plus qu'encouragés à partager ces tutoriels entre vous. Nous avons créé une classe LaTeX s'occupant pour vous de la mise en forme de votre sujet. Vous trouverez plus bas dans ce document les explications sur l'utilisation de cette classe.

Un sujet décrit un objectif que les étudiants doivent remplir dans un environnement particulier pour découvrir ou approfondir un nombre déterminé de notions. Les sujets disponibles actuellement datent de différentes époques et peuvent être très hétérogènes dans leur structure. Bien que cela ajoute de la variété, nous vous demandons plutôt d'essayer de respecter la structure suivante dans la mesure du possible :

Préambule : Le préambule d'un sujet est généralement sans rapport avec le sujet en lui-même. Il s'agit ici de partager un trait d'humour (souvent discutable) ou une curiosité avec la communauté.

Introduction : L'introduction est une présentation des grandes lignes du projet. Il est apprécié de donner un peu de contexte et une idée du travail à réaliser. Ainsi en lisant ces quelques lignes, un étudiant peut avoir une vue d'ensemble des thèmes abordés.

Objectifs : Cette partie, malheureusement souvent absente, est pourtant très importante. Il s'agit ici d'expliquer l'intérêt pédagogique du projet, car au-delà de la forme de celui-ci, un projet est avant tout un prétexte à la découverte ou à l'approfondissement d'une notion ou d'un groupe de notions. Par exemple le projet **Nibbler** : sous l'apparence d'un simple jeu **Snake**, ce projet permet d'initier les étudiants à la création d'API et de plugins pour un programme en **C++**.

Consignes générales : Cette section regroupe les consignes de base d'un projet. Langues, restrictions, autorisations, compilation, etc. En gros toutes les consignes "meta".

Partie obligatoire : Partie centrale d'un sujet, la partie obligatoire décrit en détails le travail à réaliser et les conditions imposées. Tout le secret d'un bon sujet réside dans l'équilibre subtil entre être précis et laisser une part à l'interprétation et à l'imagination. Ce facteur est important puisque c'est le moteur des discussions et des confrontations lors des soutenances.

Partie bonus : Lorsqu'on a investi du temps sur un projet et que le résultat est au rendez-vous, il est naturel d'avoir envie d'aller plus loin ! La section bonus propose des ouvertures pour répondre à cette envie. Bien entendu, la partie bonus n'est accessible que si et seulement si la partie obligatoire a été réalisée entièrement et parfaitement.

Rendu et peer-évaluation : Cette section décrit les conditions et les instructions concernant le rendu et la peer-évaluation du projet.

Rédiger un sujet est un travail long qui nécessite soin et application. Vous devez impérativement résister à l'envie de rédiger un sujet baclé de quelques lignes. Ca sera autant de temps de gagné quand viendra le temps de répondre aux questions sur le forum.

Dernière remarque concernant la langue. Nous proposons des sujets en français et d'autres en anglais. L'informatique étant un domaine dans lequel l'anglais prédomine, il sera impossible d'échapper à cette langue dans votre métier. Si vous avez les capacités pour rédiger un sujet en anglais, n'hésitez pas à vous lancer !

II.2 Le barème

Un barème à 42 est un fichier [YAML](#) qui décrit les éléments à vérifier pendant la soutenance. Un barème doit donc être murement réfléchi et rédigé avec soin.

Un barème contient les informations nécessaires au bon déroulement de la soutenance, mais également les informations nécessaires à l'intranet pour exploiter le résultat de cette soutenance, à savoir l'attribution de points d'expérience pour les différents skills liés à ce projet. Les skills disponibles à 42 sont les suivants :

- Adaptation & creativity
- Algorithms & AI
- Company experience
- DB & Data
- Functional programming
- Graphics
- Group & interpersonal
- Imperative programming
- Object-oriented programming
- Organization
- Parallel computing
- Network & system administration
- Rigor
- Security
- Technology integration
- Unix
- Web

Le créateur d'un projet doit choisir **entre 2 et 4 skills** associées à ce projet. Ce seront bien entendu les skills qui se verront attribuer des points d'xp une fois le projet validé par un étudiant. Pour cela, chaque "question" d'un barème est lié à une ou plusieurs skills des 2 à 4 skills choisies pour ce projet.

Bien entendu, l'équipe pédagogique se réserve le droit de modifier votre choix de skills ou de points dans le cas où nous ne les trouvons pas adaptés.

Nous mettons à votre disposition un script permettant de valider le format de votre barème et la somme des points de skills engagés. Nous verrons cela en détails plus bas dans ce document.

II.3 La vidéo

La vidéo associée à un projet sert de point d'entrée sur le projet aux étudiants ne connaissant rien des notions abordées sur ce projet. Enregistrer une vidéo peut sembler fastidieux et intimidant au premier abord, mais cette tâche fait partie intégrante de la création d'un projet et c'est finalement toujours un bon moment. Vous pourrez enregistrer votre vidéo avec l'aide de l'équipe **Studio42** quand le moment sera venu. Apporter une sucrerie à Mathieu Trentin le jour de l'enregistrement est toujours apprécié :). De plus, un membre de l'équipe pédagogique sera présent avec vous pour vous coacher et vous guider pendant l'enregistrement. Vous ne serez pas seul.

Tout comme les sujets et les barèmes créés par le staff de l'école, les vidéos existent dans toutes les tailles et dans toutes les qualités. Pour vous aider dans la création de votre vidéo et dans un souci d'uniformisation, je vous conseille très fortement de respecter les quelques consignes suivantes :

- Préparez les codes d'exemple que vous souhaitez utiliser dans votre vidéo à l'avance. Les silences pendant lesquels vous écrivez du code sont désagréables pour votre public.
- N'apprenez pas de texte par coeur ! La récitation rend une vidéo insupportable. Y aller "au feeling" fonctionne très bien et rend la vidéo plus naturelle.
- Une vidéo ne devrait durer que quelques minutes. Entre 5 et 10 minutes me paraît un excellent compromis.
- Utilisez le plan suivant :
 - Dites bonjour et présentez-vous.
 - Expliquez à quel projet cette vidéo est associée en rappelant brièvement le but du projet et les notions abordées.
 - Annoncez le plan de cette vidéo en trois parties : démonstration de votre implémentation, présentation des éléments techniques pertinents, puis présentation de quelques documentations ou articles en rapport avec le projet.
 - Première partie, faites une démonstration de votre implémentation du projet pour donner une meilleure idée aux étudiants du résultat attendu.
 - Deuxième partie, présentez le ou les points techniques relatifs à votre projet. Soyez précis dans votre vocabulaire, mais surtout n'en dites pas trop. Utilisez vos codes d'exemple pour aller à l'essentiel si nécessaire, mais comprenez bien qu'il ne s'agit pas ici de démontrer comment résoudre le problème. À 42, on ne donne pas les solutions !
 - Troisième partie, présentez un certain nombre de documentations et/ou d'articles qui vont ont aidé pour réaliser votre implémentation.
 - Concluez en résumant très brièvement le contenu de votre vidéo et dites un petit mot d'encouragement avant de dire au revoir.

Chapitre III

Le processus de création d'un nouveau projet

Maintenant que nous avons défini les différents éléments qui composent un projet, nous allons nous pencher sur le processus de création d'un nouveau projet.

Plusieurs fois dans ce chapitre il sera fait mention de la “plateforme de peer-pédagogie”. Cette plateforme et son utilisation seront décrites en détails plus bas dans ce document. En attendant, il suffit de savoir qu'il s'agit d'un ensemble de dépôts privés hébergés sur `GitHub` que vous devrez forker.

Pour créer un nouveau projet pour la communauté 42, il vous suffit de suivre les 10 étapes suivantes (je vais écrire "je" parce que ça fait étrange de parler de moi à la troisième personne – Thor) :

1. Envoyez-moi un mail (thor@staff.42.fr), ou passez me voir au bocal, pour obtenir un rendez-vous.
2. Le jour du rendez-vous, présentez moi votre idée et les technologies associées. Au cours de cette discussion nous adapterons votre idée au contexte de 42 jusqu'à ce que nous soyons tous les deux d'accords.
3. Faites le projet tel que nous nous sommes entendus.
4. Reprenez rendez-vous avec moi, par mail ou de visu, pour me présenter votre travail une fois celui-ci terminé.
5. Le jour du rendez-vous, nous discuterons des probables modifications qu'une implémentation réelle vous aura fait réaliser. Suivant votre résultat, retour à l'étape 2. ou passage à l'étape suivante.
6. Envoyez un mail a Quentin Perez (linus@staff.42.fr) en me mettant en copie (thor@staff.42.fr) pour demander à être ajouté sur la plateforme de peer-pédagogie. Si votre compte respecte les conditions décrites dans la section de ce document dédiée à la plateforme de peer-pédagogie, vous y aurez alors accès.
7. Rédigez votre sujet en `LaTeX` et votre barème en `YAML` puis faites une pull-request. À partir de là, deux cas possibles : soit nous acceptons votre pull-request, passez à

l'étape suivante, soit nous la refusons. Dans ce cas, le refus sera accompagné d'une justification et vous devrez recommencer cette étape.

8. Prenez rendez-vous avec Mathieu Trentin pour enregistrer votre vidéo en me mettant en copie du mail.
9. Enregistrez la vidéo.
10. Félicitations c'est terminé, vous pouvez proposer un nouveau projet! \o/

Chapitre IV

La plateforme de peer-pédagogie

L'équipe pédagogique utilise de nombreux dépôts `Git` pour stocker et versionner les sources des projets de l'école. Pour des raisons évidentes de sécurité nous ne pouvons bien entendu pas donner accès à tous nos étudiants à tous nos dépôts.

À la place, notre équipe, et en particulier Quentin Perez, a mis en place une plateforme de peer-pédagogie hébergée par `Github`.

IV.1 Prérequis

Comme mentionné dans l'étape 6 du chapitre précédent, votre compte `Github` doit respecter certaines conditions pour fonctionner avec notre plateforme.

- Connectez vous sur `github` et rendez vous sur votre [profil](#). Dans la section `Name` changez votre doux pseudo par la concaténation de votre prénom et de votre nom (`PrenomNom`) et cliquez sur `Update Profile`. Nous n'accepterons que des comptes permettant d'identifier clairement leur propriétaire. Inutile de demander à utiliser votre pseudo, la réponse est non.
- Vérifiez que vous avez un dépôt privé disponible [ici](#). `Github` offre à tous nos étudiants la formule `Micro`, si vous n'avez pas de dépôts privés, faites-en la demande [ici](#). Pensez bien à utiliser votre mail `<login>@student.42.fr` au moment de la souscription pour que `Github` accepte.
- Pour finir activez la double authentification sur `Github` [ici](#).
- Vous pouvez maintenant terminer l'étape 6 du chapitre précédent.

IV.2 Authentification

Github utilise un système de clef RSA pour authentifier ses utilisateurs. Voici donc un tutoriel pour créer votre clef. Les lignes suivantes commençant par \$ sont les commandes que vous devez taper et les informations entre [] sont les instructions à suivre.

```
$ cd ~/.ssh
$ ssh-keygen -t rsa -C "<login>@student.42.fr"
# Creates a new ssh key, using the provided email as a label
# Generating public/private rsa key pair.
# Enter file in which to save the key (/Users/you/.ssh/id_rsa): [taper "42School"]
# Enter passphrase (empty for no passphrase): [taper un mot de passe de votre choix]
# Enter same passphrase again:
$ ssh-add ~/.ssh/42School
$ pbcopy < ~/.ssh/42School.pub
```

Maintenant que vous avez généré votre clef, il faut l'ajouter sur votre profil Github [ici](#).

- Cliquez sur Add SSH key et dans la section title tapez 42School.
- Dans la partie key, clic droit -> coller.
- Puis appuyez sur Add key.

IV.3 Utilisation de la plateforme

Une fois l'étape 6 terminée, vous faites parti de l'organisation 42School sur Github. Rendez-vous [sur la page de 42School](#). Vous y découvrirez 3 dépôts privés :

projects : Ce dépôt est le dépôt principal de notre plateforme de peer-pédagogie. C'est ici que se trouve le travail déjà réalisé par d'autres étudiants avant vous, et c'est également ici que se trouvera bientôt le vôtre.

scale_validator : Ce dépôt contient le script vous permettant de valider le format de votre barème. Le dépôt scale_validator est un [submodule](#) du dépôt principal projects. Vous n'aurez donc jamais à interagir avec directement.

template : Ce dépôt contient tout le nécessaire dont vous aurez besoin pour rédiger votre sujet en LaTeX. Le dépôt template est un [submodule](#) du dépôt principal projects. Vous n'aurez donc jamais à interagir avec directement.

IV.3.1 Le dépôt projects

Le dépôt **projects** est le dépôt principal et vous devrez l'utiliser d'une manière qui vous est probablement peu ou pas familière mais qui est pourtant la manière qu'il faudrait utiliser sur **Github**. Vos droits sont limités et si vous le clonez, vous ne pourrez pas pusher vos modifications. Pourquoi ? Tout simplement pour que nous puissions contrôler ce qui se retrouve effectivement sur ce dépôt. À la place, vous allez découvrir la méthode du **fork** / **pull request**.

- Connectez vous sur github et rendez vous sur la [page](#) du dépôt **projects**.
- **Forkez** ce dépôt à l'aide du bouton en haut à droite de la page, en choisissant votre compte. **ATTENTION**, forcez-le en privé !
- Clonez **votre** fork du dépôt **project**. C'est-à-dire celui qui se trouve sur **votre** profil. Si vous vous trompez et que vous clonez le nôtre, vous ne pourrez jamais pusher.
- Créez une [branche](#) sur votre clone. Les étudiants de 42 n'utilisent malheureusement que trop peu les branches **Git** qui sont pourtant une des features les plus puissantes de cet outil.

```
$git clone git@github.com:<PrenomNom>/projects.git  
$git checkout -b add-<le_nom_de_votre_sujet>
```

Puis pour pusher votre travail quand le moment sera venu :

```
$git push origin add-<le_nom_de_votre_projet>
```

- Créez un dossier `le_nom_de_votre_projet/` à la racine du dépôt.
- Bien entendu, n'oubliez pas les deux submodules :

```
$git submodule init  
$git submodule update
```

Une fois que vous aurez terminé et pushé votre travail sur votre dépôt, il ne vous restera plus qu'à faire une **pull request** via l'interface de **Github**. Nous recevrons alors une notification. Voir l'étape 7. de la procédure.

IV.3.2 Le submodule **templates**

Pour utiliser convenablement LaTeX et notre classe de style, vous devez impérativement vous assurer que vous avez installé la distribution LaTeX pour votre système (MacTeX sous OSX), ainsi que le paquet python "Pygments" (`pip install pygments`).

Le submodule **templates** contient deux dossiers :

latex : Ici se trouve notre classe de style et les assets associés. Vous n'aurez jamais besoin d'y aller.

sample : Ce dossier contient un **Makefile** et un fichier **sample.tex** d'exemple. Copiez-les simplement dans votre répertoire de travail et renommez **sample.tex** avec le nom qui vous plaira. Ce fichier contient déjà une structure adaptée ainsi que de nombreux exemples.

Si vous avez suivi ces instructions à la lettre, un simple appel à **make** construira le pdf de votre sujet.

IV.3.3 Le submodule **scale_validator**

Pour nous assurer que les barèmes que nous uploadons sur l'intranet sont correctement formatés, Laurie Mézard a écrit un script ruby permettant de valider la structure d'un barème écrit en YAML et de s'assurer que la somme des points est cohérente. Vous trouverez trois fichiers sur ce dépôt :

scale_validator.rb : Le script ruby en lui-même. Pour le lancer il suffit de taper `ruby ./scale_validator.rb <scale.yml>`.

scale_schema.yaml : Description du format d'un barème utilisé par le script. Je vous recommande chaudement de le lire pour vous faire une bonne idée de la structure d'un barème.

sample.yml : Un barème d'exemple avec la structure de base. Il vous suffit de copier ce fichier dans votre répertoire de travail et de l'éditer.

Quelques remarques supplémentaires :

- Les champs **is_primary**, **correction_number** et **duration** ne vous concernent pas. Laissez les valeurs par défaut.
- La somme des points d'xp pour une skill particulière doit être égale à 100 parmi toutes les questions dont le type est **"standard"**.
- La somme des points d'xp pour une skill particulière doit être égale à 25 parmi toutes les questions dont le type est **"bonus"**.
- Insistez particulièrement sur les aspects fonctionnels d'un rendu. Je veux dire par là "est-ce que ceci ou cela fonctionne comme attendu?". L'évaluation de l'implémentation doit rester secondaire, voire absente dans l'idéal.

- Il est important, bien que difficile, d'être précis et exhaustif lorsque vous rédigez un barème. Gardez en tête que des étudiants n'ayant pas fait le projet sont susceptibles de le corriger !

Chapitre V

Conclusion

Vous savez maintenant tout concernant la plateforme de peer-pédagogie ! Pour toute question concernant l'utilisation de notre `GitHub` et de ses dépôts, adressez vos questions à Quentin Perez (`linus@staff.42.fr`). Pour toute question concernant le script de validation des barèmes, adressez vos questions à Laurie Mezard (`laurie@staff.42.fr`). Dans tous les cas, mettez-moi en copie (`thor@staff.42.fr`). Enfin pour toute autre question adressez-les moi.

Je vous souhaite bon courage dans la création de votre projet, et je vous remercie d'y consacrer du temps.