

Sujet de stage M1 Info

« Développement full-stack : industrialisation d'un proof-of-concept
pour learn-ocaml »

Encadrant : Érik Martin-Dorel (erik.martin-dorel@irit.fr)

Janvier 2022

1 Informations pratiques

- Stage de M1 Info à temps plein
- Nature du travail : développement logiciel/web
- Proposition de dates : début mai → fin juillet (3 mois)
- Gratification de stage (taux horaire en vigueur depuis 2020) : 600,60 €/mois
- Établissement : [IRIT](http://irit.fr)
- Contact : erik.martin-dorel@irit.fr (joindre CV+LM pour candidater)

2 Description du contexte du sujet

Depuis l'année 2016-2017, l'UE PFITA (Programmation Fonctionnelle et Introduction aux Types Abstraits) de la Licence 3 Informatique de l'Univ. Paul Sabatier utilise un outil d'évaluation automatique à base de tests unitaires et d'analyse statique, et du moteur de MOOC learn-ocaml¹ développé par la fondation OCaml. Cet outil génère un rapport HTML indiquant les erreurs ou les motifs de code à éviter.

Depuis 2017, nous avons développé dans le projet PFITAXEL (auquel ont participé 10 étudiants de L3 ou M1 de l'UPS) et en collaboration avec la fondation OCaml :

- une seconde application web² pour faciliter le travail des enseignants dans la conception de nouveaux sujets de devoir maison et de TP ;
- un front-end Emacs³ pour accéder en tant qu'étudiant aux fonctionnalités de la plateforme learn-ocaml directement à partir d'un IDE OCaml standard de type « Tuareg » ;
- une extension de learn-ocaml permettant de gérer l'authentification par le protocole LTI⁴ ou login/mot-de-passe (en cours d'intégration).

3 Proposition de sujet de stage

Un précédent stage (L3) a permis la conception d'une fonctionnalité clé pour Learn-OCaml (« multi-part exercises ») et l'écriture d'un premier PoC.

1. <https://github.com/ocaml-sf/learn-ocaml>

2. learn-ocaml-editor : <https://pfitaxel.github.io/pfitaxel-demo/>

3. learn-ocaml.el : <https://melpa.org/#/learn-ocaml>

4. https://en.wikipedia.org/wiki/Learning_Tools_Interoperability

Cette fonctionnalité⁵ consiste à ajouter à Learn-OCaml, de manière *backward-compatible*, un nouveau type d'exercice regroupant de manière ergonomique plusieurs exercices individuels, correspondant typiquement à plusieurs parties d'un même problème. La fonctionnalité permettra aussi de définir des méta-données spécifiques telles que la stratégie d'évaluation de l'exercice global.

Le but de ce stage est de poursuivre cet effort de développement :

- en finalisant l'implantation de cette User Story, côté back-end (OCaml) et front-end (HTML5/CSS/js_of_ocaml)
- en co-développant avec Erik Martin-Dorel l'évolution requise côté front-end Lisp (learn-ocaml-mode)
- et en aidant à l'intégration de cette fonctionnalité dans le dépôt maître.

Le code développé sera placé sous une licence libre permissive (MIT).

Le stage s'effectuera en contexte agile, en collaboration étroite avec Yann Régis-Gianas, co-mainteneur du projet Learn-OCaml, ainsi que les autres étudiants recrutés.

4 Environnement technologique

Environnement de développement : [Git](#) et [GitHub Actions](#), [GNU Emacs](#), [Tuareg](#) et [Merlin](#), [Docker](#)

Langages : [OCaml](#), [JavaScript](#), [Emacs Lisp](#)

Frameworks : [Lwt](#), [js_of_ocaml](#)

5 Pré-requis

Sont attendus : un intérêt prononcé pour la programmation fonctionnelle, une bonne expérience du langage OCaml (en particulier de son fragment fonctionnel pur), des connaissances solides en génie logiciel, en gestion de version décentralisée (Git) et en Web Services (REST), et de bonnes aptitudes de rigueur et de synthèse.

Non demandées : expérience préalable en virtualisation applicative avec Docker, ou connaissance des frameworks impliqués (Lwt, js_of_ocaml).

5. Décrite dans l'issue [#331](#).