



OCaml PRO

Améliorations pour l'application web "Try OCaml"

Stage 3-6 mois, niveau M1 à M2 Recherche.

Présentation d'OCamlPro :

OCamlPro a été fondé en 2011 par d'anciens chercheurs d'Inria avec pour mission d'aider les utilisateurs industriels à adopter OCaml, le langage de programmation de pointe. Nous concevons et mettons en œuvre des logiciels à forte valeur ajoutée pour nos clients et nous avons une longue expérience dans le développement et la maintenance d'outils open source pour OCaml.

Présentation de Try OCaml :

[Try OCaml](#) est un portail web offrant un éditeur simple et la possibilité d'évaluer son code OCaml directement depuis le navigateur. C'est un outil précieux pour abaisser la barrière d'entrée vers le langage, vu qu'il ne requiert aucune installation et permet aussi bien à des nouveaux arrivants de tester le langage qu'à des étudiants de réaliser leurs exercices, ou même à des développeurs de lancer des tests rapides.

La première version de Try OCaml, née en 2011, qui offrait un interprète/compilateur en ligne mais pas de fonctions d'édition, a été remplacée cette année par une version plus avancée munie d'une interface au goût du jour. Cette version réutilise pour son moteur une partie du travail accompli pour le projet [Learn-OCaml](#).

Sujet de stage :

Fonctionnelle et plébiscitée par la communauté OCaml, cette nouvelle version est cependant toujours en version "Beta", et nous aimerions y apporter une gamme d'améliorations de petite et plus grande envergure. Les objectifs pourront être choisis avec le candidat dans la liste suivante, suivant son niveau, la durée du stage et ses ambitions :

1. Améliorations d'interface :

- Meilleure gestion des longs espacements lors de l'envoi du code au toplevel
- Découpage automatique des définitions racines (inspiré de <https://github.com/ocamlpro/ocaml-top>). Cela permettrait d'offrir une interface collant plus près au langage, et éviterait des erreurs de programmation récurrentes liées aux `in`, `;` et `;;`.
- Ajout de raccourcis clavier et de fonctions d'édition plus avancées

2. Fonctionnalités OCaml :

- Ajout de bibliothèques graphiques et affichage dans la page
- Fournir un catalogue de bibliothèques à charger dynamiquement
- Proposer une API pour permettre de gérer des éléments web le plus naturellement possible à partir du code OCaml

3. Fonctionnalités web :

- Offrir un moyen de partager des extraits de code via le site
- Proposer des facilités pour déployer directement des mini-applications en OCaml

Les premiers de ces éléments ne requièrent qu'une connaissance d'OCaml, une compréhension de ``js_of_ocaml`` et des aspects web correspondants. Il s'agira ensuite de rentrer plus avant dans les internes du logiciel, et de pouvoir manipuler la chaîne de compilation OCaml-javascript embarquée dans la page web.

Enfin, un objectif avancé, susceptible de dépasser Try OCaml lui-même, sera d'étudier les bibliothèques existantes et si besoin de définir les bonnes abstractions pour offrir les moyens de développer, directement depuis TryOCaml, des mini-applications impliquant de l'animation et de l'interaction utilisateur. On peut s'inspirer de ce qui est fait dans [le langage Elm](#).