Appel à devis : prestations Hyphe Cloud / Text

Votre dossier de candidature (devis, CV(s), commentaire technique) doit être adressé au responsable de DIME WEB benjamin.ooghe@sciencespo.fr avant le 15/01/2020.
Toute question peut être soumise à la même adresse.

Nous souhaitons un début de travaux en février 2020 et une fin au plus tard au 15/06/2020.

Sommaire de l'appel à devis

Contexte et mode de consultation

Ce document présente le contenu d'une mission de développement informatique que Sciences Po souhaite confier à un prestataire. Il s'agit plus particulièrement d'une mission commanditée par l'instrument de recherche DIME WEB hébergé au médialab et faisant partie de l'équipement DIME-SHS.

La mission consiste à améliorer l'adoption et les fonctionnalités du logiciel Hyphe en développant deux applications web :

- une application web de déploiement du logiciel Hyphe sur une infrastructure docker d'un prestataire d'hébergement Cloud ;
- un prototype d'indexation et d'analyse des données textuelles collectées dans un corpus web créé par Hyphe.

L'objet de ce document est de préciser les spécifications du logiciel afin de permettre l'obtention de devis de différents prestataires qui seront mis en concurrence. Les

spécifications laissent un certain nombre de points non détaillés. La prestation sera menée par une gestion de projet agile. L'engagement se fait donc sur un temps et non sur un forfait de réalisation.
Le devis doit en revanche présenter une estimation du temps nécessaire à la réalisation des outils décrits ainsi qu'un tarif associé et un calendrier des itérations. Les prestations devront être livrées au plus tard le 15/06/2020.
Tout élément précisant le profil du développeur qui prendrait la mission en charge et l'analyse du besoin technique seront pris en compte dans la mise en concurrence.
Seront évalués :

- l'adéquation des expériences passées des prestataires avec la demande : recherche académique, web mining, open source, application web complexe, visualisation de réseaux, docker, indexation textuelle ;
- le rapport expertise/tarif des intervenants ;
- la pertinence de l'estimation et des commentaires associés.

Les réponses peuvent adresser les deux lots décrits ou un seul.

Votre dossier de candidature (devis, CV(s), commentaire technique) doit être adressé au responsable DIME WEB benjamin.ooghe@sciencespo.fr avant le 15/01/2020.
Toute question peut être soumise à la même adresse.

Gestion de la prestation

La prestation est un engagement à du temps de développement et non à un forfait de réalisation. Cela explique que cet appel à devis se concentre plus sur les besoins fonctionnels que sur les détails techniques de la réalisation.
La prestation sera organisée suivant une méthode librement inspirée des approches agiles. À chaque itération les objets de travail, leur importance et leurs difficultés seront discutées et planifiées avec le prestataire. La conception graphique est assurée par le client qui fournit les mock-ups.
La prestation est à réaliser à distance. Le prestataire devra se rendre dans les locaux de Sciences Po pour la réunion de lancement et pour chaque démonstration de fin d'itération. Les éventuels frais de déplacement sont à inclure au devis.
L'ensemble du code produit dans le cadre de cette prestation sera livré sur le répertoire GitHub du logiciel hyphe pour le lot 2 et sur un répertoire GitHub dédié de l'organisation médialab pour le lot 1. Dans les deux cas le code produit sera ouvert sous license libre AGPL-3.0.

Lot 1 : plateforme d'assistance au déploiement Cloud de Hyphe

Le logiciel Hyphe est un outil permettant à des chercheurs ou des étudiants de créer un corpus de pages web dédié à leur sujet d'enquête. Il est possible d'installer et d'utiliser ce logiciel librement à travers le système de virtualisation Docker. Deux problèmes limitent son installation :

- l'utilisation de Docker n'est pas aisée pour les utilisateurs cibles qui sont des étudiants, professeurs et chercheurs en Sciences Humaines et Sociales ;
- le fonctionnement du logiciel demande des ressources (mémoire, processeur et espace disque) potentiellement importantes qui nécessitent l'usage d'un serveur et non d'un ordinateur personnel.

Aussi, DIME WEB souhaiterait pouvoir faciliter le déploiement du logiciel Hyphe sur une infrastructure cloud hébergée en Union Européenne à travers ses images Docker[1].

Scénario d'usage

Cette application de déploiement permettrait à un utilisateur non technicien d'automatiser le déploiement d'un serveur Hyphe. Le scénario d'usage est le suivant :

- un utilisateur se rend sur l'application car il souhaite utiliser Hyphe sans avoir à l'installer sur son ordinateur ;
- l'application doit dans un premier temps documenter le processus de déploiement : quelles sont les grandes étapes, quel sera le coût de l'hébergement et ce qu'il doit faire en amont (création d'un compte chez l'hébergeur avec moyen de paiement associé) ;
- l'application propose ensuite à l'utilisateur de commencer un déploiement en demandant un moyen d'authentification pour accéder à l'API de l'hébergeur choisi (credentials) ;
- après avoir vérifié la validité de l'authentification, l'application guide l'utilisateur dans la configuration du serveur Hyphe qu'il s'apprête à déployer. Cette étape explicite les conséquences, en termes de coût s'il y en a, des différentes options choisies[2] (les éléments configurables par l'utilisateur seront précisés par l'équipe DIME WEB) ;
- après une validation par l'utilisateur des choix de configuration et un rappel du coût, l'application lance le déploiement en dialoguant avec l'API du service d'hébergement ;
- les étapes de déploiement sont documentées à l'utilisateur avec la possibilité si il le souhaite de voir le log technique des requêtes ;
- une fois le déploiement terminé un lien vers le serveur est communiqué ;
- si l'API de l'hébergeur le permet, l'application devra fournir une information sur l'usage du serveur créé pendant son utilisation ou documenter comment trouver cette information sur le site de l'hébergeur, ainsi que sur les modalités d'arrêt et suppression du serveur.

Contraintes techniques

Afin de limiter les besoins de maintenance pour Sciences Po et si l'API de l'hébergeur le permet l'application devrait être une application web statique (sans serveur). Cela permettrait aussi de garantir que les credentials ne soient pas communiqués à un serveur

---

[1] Voir les images hyphe_backend, hyphe_crawler et hyphe_frontend disponibles sur DockerHub : https://hub.docker.com/u/scpomedialab et la documentation d'installation avec Docker : https://github.com/medialab/hyphe#easy-install-using-docker

[2] Voir les configurations suivantes :
- https://github.com/medialab/hyphe/blob/master/.env.example
- https://github.com/medialab/hyphe/blob/master/config-backend.env.example
- https://github.com/medialab/hyphe/blob/master/config-frontend.env.example

central autre que celui de l'hébergeur. Mais cela n'est possible que si l'API autorise les appels cross-origin. À défaut, l'application web pourrait être embarquée dans l'application desktop HyBro[3] (reposant sur electron) téléchargeable par les utilisateurs.

Le service d'hébergement doit permettre de choisir l'hébergement sur un datacenter localisé dans un pays de l'Union Européenne. Le choix du service d'hébergement sera fixé après discussion des faisabilités. Nous avons déjà identifié une solution basée sur une api OpenStack mais nous ne souhaitons pas la communiquer dans cet appel pour laisser place à des propositions alternatives.

Éléments indispensables du devis pour le lot 1

Toute réponse sur ce lot devra comprendre :

- un court descriptif des technologies que le prestataire propose d'utiliser ainsi qu'une esquisse de l'architecture technique ;
- la proposition d'un ou plusieurs services d'hébergement susceptibles d'accueillir des déploiements automatiques d'images docker Hyphe tel que décrit dans le scénario d'usage ;
- une estimation du temps de développement, structuré en phases de travail datées précisant les noms des intervenants.

Lot 2 : indexation et analyse des contenus textes des corpus web Hyphe

Le logiciel Hyphe permet de créer des corpus web[4] (sélection de pages web agrégées par des préfixes d'URLs) en crawlant les pages de sites Internet. Les données collectées sont les contenus HTML et les liens extraits de ce contenu. À ce jour, Hyphe ne propose pas d'analyse du contenu texte que l'on peut extraire du contenu HTML. Des expérimentations ont déjà été menées en ce sens. Ce lot consiste à créer un premier prototype intégré à Hyphe qui permette d'extraire et d'exploiter les contenus textuels d'un corpus Hyphe.

Scénarios d'usage

Lot 2.1 : Création de l'index des contenus texte

- un utilisateur instancie un nouveau corpus web avec Hyphe ;
- il choisit d'activer l'indexation des contenus textes (avant de commencer son corpus, non modifiable ensuite), auquel cas, Hyphe stocke les contenus web zippés dans une base mongoDB (déjà en place) ;
- un processus parallèle à l'application serveur de Hyphe surveille la collection contenant les pages crawlées dans la base de données mongoDB des corpus dont

[3] https://github.com/medialab/hyphe-browser
[4] Jacomy, Mathieu, Paul Girard, Benjamin Ooghe and Tommaso Venturini. 2016. "Hyphe, a Curation-Oriented Approach to Web Crawling for the Social Sciences." Poster presented at International AAAI Conference on Web and Social Media, Köln, GERMANY, May 18-20.

l'indexation est activée et récupère les contenus texte par paquets, en extrait le texte puis les indexe dans une structure mémoire ;
- le processus surveille entre chaque batch une collection mongoDB complémentaire qui liste les mises-à-jour (changements de WebEntity associée à certaines pages) et les répercute sur l'index.

Lot 2.2 : Construction d'une requête texte

- un utilisateur construit un corpus web avec Hyphe en ayant activé l'option d'indexation du texte ;
- pendant la création de son corpus il souhaite analyser le texte des pages crawlées ;
- une interface dédiée lui permet de construire une requête pour identifier les WebEntities utilisant un vocabulaire (voir "I. Search-engine-like view, example 2 corpus view" du mockup) ;
- la même interface peut être utilisée pour voir le résultat d'une requête sur les pages web d'une WebEntity particulière (voir "I. Search-engine-like view, example 1 inside a WebEntity" du mockup) ;
- ces applications permettent de chercher mais aussi et surtout de créer des requêtes appelées "issue-facet" dans le mockup et stockables pour réutilisation ultérieure dans d'autres interfaces de l'application web.

Lot 2.3 : Croisement contenu texte et réseau des WebEntities

- un utilisateur construit un corpus web avec Hyphe en ayant activé l'option indexation du texte ;
- pendant la création de son corpus il souhaite analyser le textes des pages crawlées ;
- une interface dédiée lui permet de projeter le résultat d'une requête pré-construite ("issue-facet" du lot 2.2) sur le réseau des hyperliens entre les WebEntities (fourni via l'API du serveur Hyphe) ;
- le réseau représente visuellement les résultats de la requête, ce qui permet d'interpréter la répartition structurelle d'une requête textuelle et donc les effets de "framing" au sein des communautés hypertextuelles.

Contraintes techniques

Le lot 2.1 a été prototypé dans la branche "*live_text_indexing*" du repository de Hyphe via le script python3 suivant :
https://github.com/medialab/hyphe/blob/live_text_indexing/hyphe_text_indexation/liveTextIndexation.py

Les lots 2.2 et 2.3 doivent être développés dans les interfaces et la base de code existante du logiciel hyphe. Le framework client est AngularJS 1.6. Pour les requêtes client - structure mémoire, il sera préférable de créer un serveur d'application dédié à l'usage de l'index texte (et non étendre le module CORE existant).

Les services développés devront être encapsulés dans un environnement virtuel python3 dédié et assemblés dans des images-conteneurs complétant la stack Docker actuelle de Hyphe[5].

Le prototype actuel utilise ElasticSearch comme structure mémoire d'indexation mais les prestataires peuvent proposer d'autres solutions.

Éléments indispensables du devis pour le lot 2

Toute réponse sur ce lot devra comprendre :

- un court descriptif des technologies que le prestataire propose d'utiliser ainsi qu'une esquisse de l'architecture technique ;
- une estimation du temps de développement structuré en phases de travail datées précisant les noms des intervenants.

[5] voir https://github.com/medialab/hyphe/blob/live_text_indexing/hyphe_text_indexation/Dockerfile et https://github.com/medialab/hyphe/blob/live_text_indexing/docker-compose-text-indexation.yml