

6GEI466 - Applications réseaux et sécurité informatique - Laboratoire #2

Jean-Luc Cyr

Hiver 2025

1 Objectif

Vous familiariser avec le développement en javascript et parfaire vos connaissances en python pour le web.

2 Introduction

Le but de ce laboratoire est de paufiner votre serveur développé lors du premier laboratoire et inclure des composantes javascript dans votre interface web.

Pour ce faire, vous devez modifier votre serveur web (HTTP) en langage python pour y ajouter une nouvelle route (URL) qui permet de récupérer des données en format JSON ou XML. De plus dans votre page web vous devrez inclure la bibliothèque JQuery pour faire un appel AJAX et manipuler le DOM (Structure HTML) de votre page.

Le tout sera mis en place dans un environnement virtuel.

3 Installation de l'environnement de travail

Pour réaliser votre laboratoire, je vous conseille d'utiliser une version récente de python (3.12+ si possible).

Pour débiter créez un environnement virtuel (de type venv).

Exemple de création et activation d'un environnement virtuel.

```
python -mvenv <mon_repertoire>
cd <mon_repertoire>
scripts\activate
```

Par la suite vous pourrez installer la bibliothèque Flask et générer le fichier "requirement.txt" de votre environnement.

```
pip install flask
pip freeze >requirement.txt
```

L'auxiliaire vous montrera au besoin lors de la séance de laboratoire comment initialiser un environnement virtuel plus en détail et vous accompagnera pour le faire.

4 Description

En utilisant le serveur en langage python réalisé avec la bibliothèque flask modifiez le pour :

4.1 Votre serveur devra

- Recevoir des requêtes HTTP
- Les appels dont l'adresse commence par /static/* devront retourner le fichier correspondant dans le répertoire static sur le disque
- Les appels à l'adresse / devront être traités par le logiciel et retourner le contenu d'un gabarit nommé accueil.html
- L'appel à l'adresse /horoscope devra retourner des données (texte, xml ou json) contenant pour une date précise reçu le signe du zodiac correspondant, le nom d'un fichier image correspondant et un court texte de prédiction astrale.
- Les appels aux autres pages devront retourner une erreur 404 et le contenu d'un gabarit nommé erreur404.tpl

4.2 Le fichier HTML de la page d'accueil devra contenir les éléments suivants

- Les éléments d'une page HTML 5 minimale valide
- Un titre <h1> (dont la valeur est mise en place par le logiciel)
- Un formulaire <form>
- Un champ nom et un champ prénom.
- un champ date (utiliser l'affichage d'un calendrier "date picker" de jqueryui ou une autre validation)
- un bouton voir mon horoscope
- une zone (div) qui affichera l'horoscope.
- Un logo quelconque (image jpg ou autre)

4.3 Comportement

Lorsque la page racine (/) est invoquée elle doit afficher le formulaire (sans horoscope) Si les champs sont remplis et que le bouton "voir mon horoscope" est pressé : Vous devez transmettre les information à votre serveur à l'aide d'une requête AJAX (POST) avec les valeurs de nom, prénom et la date. L'URL utilisé pour la requête doit être : /horoscope Vous devez afficher le résultat de la requête AJAX dans la zone destiné à afficher l'horoscope.

Lorsque la page du service “horoscope” (/horoscope) est appelée avec la méthode POST Vous devez valider que vous avez reçu un nom, prénom et une date Si un paramètre est manquant ou vide, vous devez retourner une erreur avec le message paramètre manquant. Vous devez valider la date reçue Si elle est invalide, retourner un message “date invalide” Si elle est valide, retourner un message contenant “Le nom, prénom, le signe du zodiaque (idéalement accompagné d’un lien <a> vers son image) correspondant à la date accompagné d’un court texte de votre choix prédisant l’horoscope.”

Tous les appels non prise en charge par le système (autre que /static/* et /) doivent retourner un erreur 404 avec le contenu du gabarit erreur404.html

Vous pouvez faire votre propre @route et fonction ou utiliser les outils fournis par flask pour gérer les pages inexistantes à votre guise.

5 Politique de remise de travail

Méthodes de remise

1. via le “Moodle” du cours, Les fichiers devraient être regroupés dans un fichier “zip” ou “tar” afin de préserver l’arborescence du projet.

Remettez seulement les fichiers de votre programme et non pas tout l’environnement virtuel (donc pas les sous répertoires scripts, include, lib, etc).

Les fichiers

- .py (vos programmes)
- requirement.txt (vos dépendances)
- readme.txt (les informations sur la version de python, les instructions de mise en route de votre programme)
- votre répertoire “static” et son contenu (images, css, etc)
- votre répertoire “templates” et son contenu (vos pages/templates.html)

Attention, la remise doit avoir lieu avant la date inscrite sur le site du cours. Toute remise après cette heure de tombé sera considéré comme remise en retard et soumis à la pénalité de 10%/jour et ne sera pas corrigé après 7 jours de retard. Prévoyez les problèmes et délais possible de transmission internet, n’attendez pas à la dernière minute.

6 Correction

Le laboratoire vaut 10% de la note du cours.

La correction sera effectuée sur les éléments suivants :

- o Suivi des consignes (2 points)
 - . structure du projet
 - . séparation du contenu dans les fichiers (python, html, css, js, ...)
 - . structure des répertoires (static, template, ...)
 - . fichier d’instructions readme.txt
 - . fichier de dépendance requirements.txt

- o Fonctionnalités (5 points)
 - . Recevoir des requêtes HTTP (GET et POST)
 - . Recevoir la requêtes HTTP AJAX (POST) sur /horoscope et retourner du data (JSON, XML ou TEXTE)
 - . Traiter la réponse de la requête AJAX correctement (Sécurité, Erreurs, Données) et faire l’affichage approprié
 - . Les appels dont l’adresse commence par /static/* devront retourner le fichier correspondant dans le répertoire static sur le disque
 - . Les appels à l’adresse / devront être traités par le logiciel et retourner le contenu d’un gabarit nommé accueil.html
 - . les balises demandées sont utilisées correctement
 - . affichage des informations (+ sécurité = validation des données sur-tout la date)
 - . Les appels aux autres pages devront retourner une erreur 404 et le contenu d’un gabarit nommé erreur404.html
- o Validité du code et des documents (3 points)
 - . Validation HTML/CSS (avec l’outil du w3c validator)
 - . Documentation (explication dans le fichier readme.txt comment démarrer votre serveur et l’adresse pour y accéder)
 - . Commentaires dans le code