Document et Web Sémantique - TP Production d'un site Web

L'objectif de ce TP est d'utiliser le CI/CD de gitlab pour créer un mini site Web qui présente les contrats pédagogiques du département ITI.

Préambule

Ce TP nécessite la configuration suivante d'un projet gitlab :

- Créez un projet gitlab (que vous supprimerez à la fin du semestre) en suivant la nomenclature suivante : `ITI4_DOCU_PRENOM_NOM_20XX_YY` (ou `XX` et `YY` représentent l'année universitaire par exemple `24_25`)
- Activez l'intégration continue pour la projet (Paramètres > Général > Visibilité, fonctionnalités du projet, autorisations > CI/CD et surtout enregistrer) ;
- Ajoutez, grâce l'éditeur gitlab (Compilation > Éditeur de pipeline) ou depuis votre éditeur, un fichier `.gitlab-ci.yml` à la racine de votre branche, tel que :
  - l'image sera `debian:bullseye-slim` :

    ```
    image: debian:bullseye-slim
    ```

  - il y aura, pour l'instant, une seule étape, `deploy`, dans votre CI/CD :

    ```
    stages:
      - deploy
    ```

  - il y aura l'installation des paquets `default-jre libxml2-utils` et `libsaxonb-java` pour pouvoir utiliser `xmllint` et `saxonb-xslt` :

    ```
    before_script:
      - apt-get update && apt-get install -y default-jre libxml2-utils
          libsaxonb-java
    ```

  - le job `pages` associé à l'étape `deploy` produira des pages html dans le répertoire `public` (cet *artefact* aura une durée de vie de 7 jours) :

    ```
    pages:
      stage: deploy
      script:
        - mkdir public
        - mkdir public/fr
        - mkdir public/en
        - echo "<html><head><title>ITI</title></head><body><a href='fr/index.
            html'>Fr</a><br/><a href='en/index.html'>En</a></body></html>" >
            public/index.html
      artifacts:
        expire_in: 7 days
        paths:
          - public
    ```

— L'URL du site Web produit (une fois que le pipeline CI/CD s'est déroulé sans problème) est donné par la page Déploiement > Pages

1 Feuille XSL-T `ec.xsl`

Complétez la feuille `xslt/ec.xsl` qui produit la page HTML de description d'un EC identifié à grâce au paramètre `idEc`. La langue de cette page sera donnée par le paramètre `langue` qui par défaut vaudra `'fr'`.

La figure 1 montre un exemple de rendu de la page HTML présentant l'EC Algo en français.

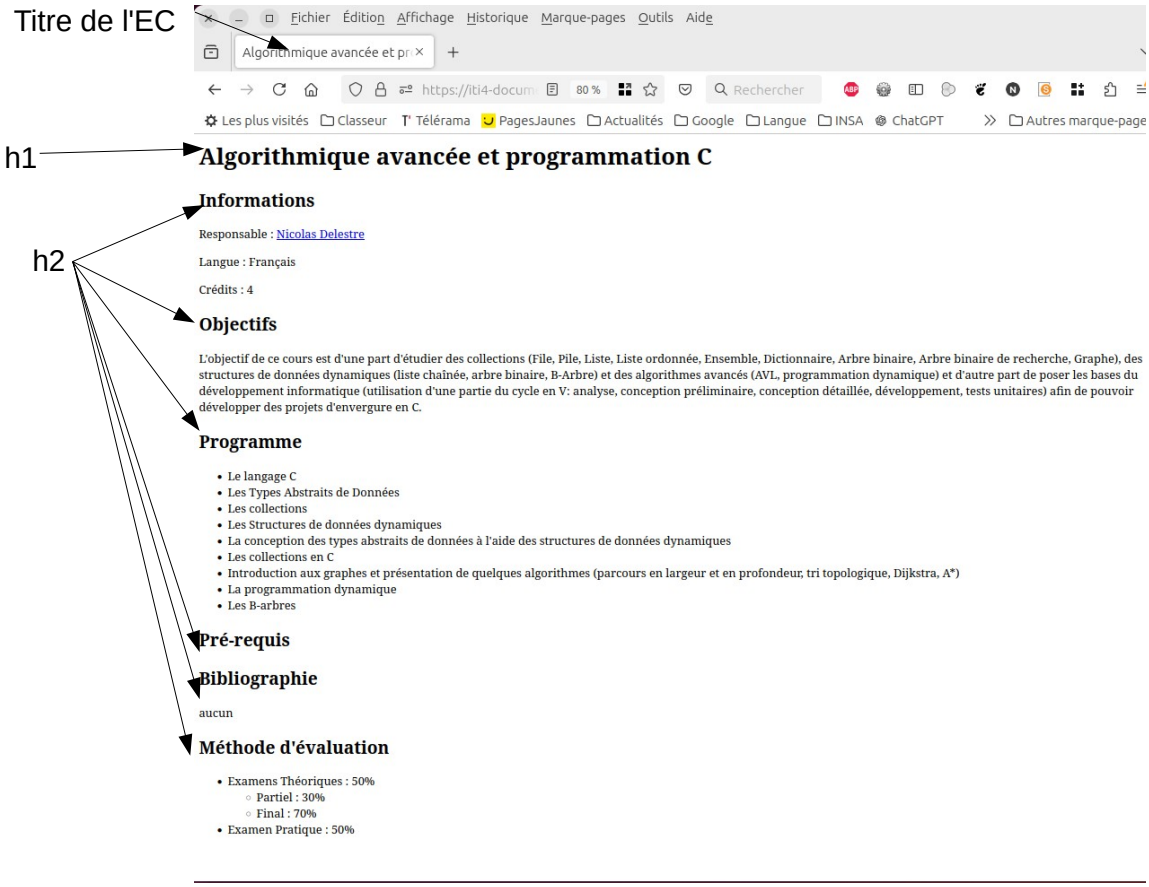

FIGURE 1 – Page HTML d'une EC

2 Génération des pages HTML des EC

La commande suivante permet d'obtenir l'attribut `id` de toutes les EC :

```
xmllint --noent --xpath "/specialite/ecs/ec/@id" xml/iti.xml
```

Pour extraire uniquement les identifiants, on peut utiliser l'éditeur en ligne `sed` pour extraire uniquement les valeurs de ces identifiants :

```
sed 's/^ id="\([^"]*\)"/\1/'
```

Ajoutez au job `pages` le code `bash` permettant de générer les fichiers `html` pour chaque EC (le nom du fichier sera celui de l'`id`) dans le répertoire `fr` pour le français et dans le répertoire `en` pour l'anglais.

3 Feuille XSLT `contrat.xsl`

Complétez la feuille `xslt/contrat.xsl` qui produit la page HTML de description d'un contrat (d'un semestre) identifié à grâce au paramètre `semestre`. La langue de cette page sera donnée par le paramètre langue qui par défaut vaudra `'fr'`.

La figure 2 montre un exemple de rendu de la page HTML présentant l'EC Algo.

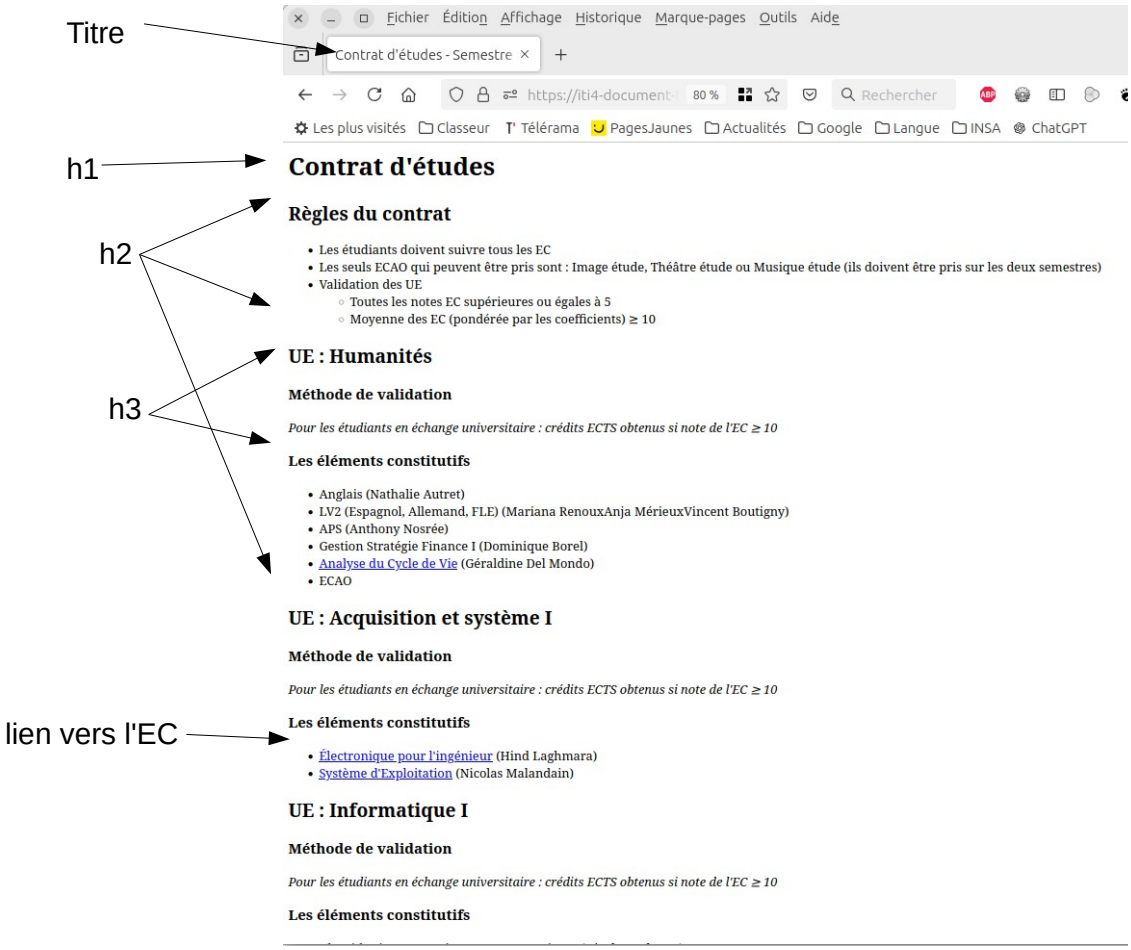

FIGURE 2 – Page HTML d'un contrat

4 Génération des pages HTML des contrats

La commande suivante permet d'obtenir l'attribut `id` de toutes les EC :

```
xmllint --noent --xpath "/specialite/contrats/contrat/@semestre" xml/iti.xml
```

Ajoutez au job pages le code `bash` permettant de générer les fichiers `html` pour chaque contrat (le nom du fichier sera `'semestre'` suivi l'`id`) dans le répertoire `fr` pour le français et dans le répertoire `en` pour l'anglais.

5 Feuille XSLT `contrats.xsl`

Complétez dans la feuille `xslt/contrats.xsl` qui produit la page HTML permettant d'accéder aux contrats. La langue de cette page sera donnée par le paramètre langue qui par défaut vaudra `'fr'`.

La figure 3 montre un exemple de rendu de la page HTML présentant l'EC Algo.

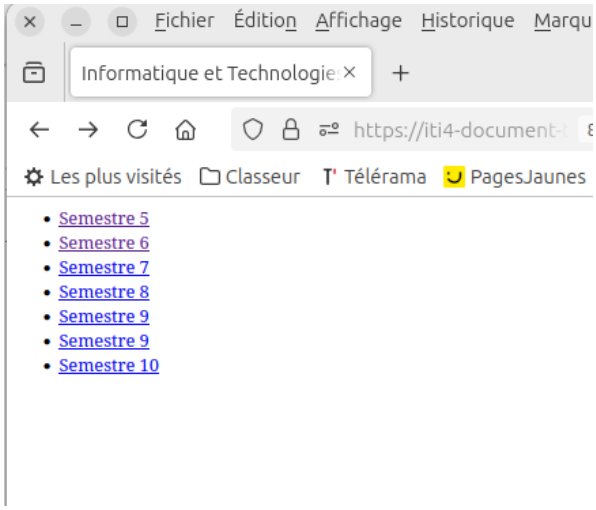

FIGURE 3 – Page HTML des contrats

6 Génération des pages HTML d'une spécialité

Ajoutez au job pages le code `bash` permettant de générer la page `index.html` d'une spécialité pour le français et l'anglais.

Annexe

Vous trouverez dans le répertoire `xsl`, le fichier `traduction.xml` qui contient les traductions des titres de chaque section des documents à générer. Depuis une feuille XSL-T, vous pouvez utiliser ce document à l'aide de la fonction xpath `document`. Par exemple la déclaration de la variable XSL-T suivante permet d'utiliser ce document XML :

```
<xsl:variable name="traduction"
              select="document('traduction.xml')//vocabulaire"/>
```

Ainsi si dans une feuille XSLT-T on a besoin de produire la traduction de "contrat d'études" en anglais, on utilisera le code XSL-T suivant :

```
<xsl:value-of select="$traduction/intitule-contrats-etudes[lang('en')]/>
```