

Unités de formation « Projet de développement SGBD » et « Projet de développement Web »

Portfolio d'évaluation de la première partie

Document de présentation du cahier des charges, de l'environnement de développement et de l'évaluation pour le travail servant de base à l'évaluation de la première partie du « portfolio »

Cahier des charges

Sujet

Vous devez développer un sous ensemble d'une application de gestion d'écoles de « Promotion Sociale », plus spécifiquement la gestion des personnes (étudiants, enseignants, personnel administratif), des sections, des unités d'enseignement (ue) et leurs capacités, et des ue « organisées ».

Une page web sera réalisée pour chacun des points mentionnés ci-dessus, à l'exception des capacités qui seront gérées dans la même page que les ue.

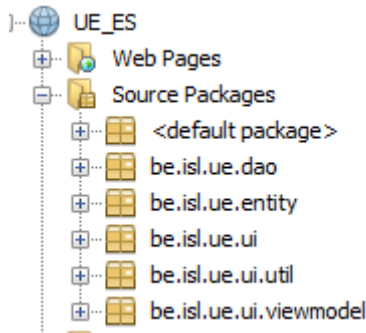
L'application sera développée dans le langage Java SE pour la partie « Dev SGBD » et Java SE (+ Servlet), HTML, CSS, JavaScript pour la partie « Dev Web ». L'utilisation de Bootstrap est tolérée, **aucune autre librairie tierce ne peut être utilisée.**

L'application sera réalisée en utilisant à bon escient les technologies vues aux cours.

Les données seront stockées dans la base de données « ue » PostgreSQL se trouvant sur le serveur Titus de l'Institut.

Structure du projet principal

Le code de l'application sera organisé en « package » (entity, dao, ui,...).



Package Entity (be.isl.ue.entity)

Contiendra les classes entity.

Package Business (be.isl.ue.dao)

Contiendra les DAO. Chaque classe DAO contiendra, au moins, les méthodes suivantes :

- List<E> load() : Charger et retournera une liste d'entity du type de DAO.
- List<E> load(ViewModel) : Charger et retournera une liste d'entity correspondant aux critères se trouvant dans le view model.
- List<E> getList() : retournera la liste des entity chargés par une des méthodes load.
- void save(E) : effectuera la sauvegarde de l'entity passé en paramètre. Si l'id de l'entity est null, un insert dans la DB sera effectué, sinon un update sera effectué.
- void delete(E) : supprimera de la DB l'entity passé en paramètre.

Package UI (be.isl.ue.ui)

Contiendra les servlets => un servlet par page principale à gérer. Le premier paramètre passé au servlet, correspondra à l'action à réaliser. Le nom des actions sera standardisé au maximum entre les différentes pages (par exemple : 'search', 'save', 'delete', ...)

Package util (be.isl.ue.ui.util)

Contiendra les éventuelles classes utilitaires pouvant aider à la gestion de l'interface utilisateur.

Package viewmodel (be.isl.ue.ui.viewmodel) (ou be.isl.ue.vm)

Contiendra les éventuelles classes modèles correspondantes aux vues utilisateurs.

Fichiers « web »

Une page html par domaine à gérer :

- index.html
- person.html
- section.html

- ue.html

Un fichier CSS pour tout le site.

Un fichier JavaScript commun (fonctions utilitaires), et un fichier par page :

- Person.js
- Section.js
- Ue.js
- Capacity.js

Interface utilisateur

L'interface utilisateur sera divisée en 5 parties comme illustré dans le schéma ci-dessous.

La zone de « Titre » servira à afficher le titre de la page courante.

La zone « Menu » affichera le menu composé de liens permettant d'accéder aux différentes pages de l'application.

La zone « Critères de Recherches » sera constituée d'un formulaire permettant à l'utilisateur d'encoder ses critères de recherches, qui seront utilisés pour construire une requête « SQL » et ainsi interroger la base de données.

La zone « Liste Résultats » affichera un tableau reprenant en lignes les enregistrements correspondant aux critères de recherches, et en colonnes les informations les plus pertinentes permettant à l'utilisateur d'identifier une ligne. Chaque ligne de ce tableau comportera un hyperlien qui permettra d'afficher le détail de l'élément sélectionné.

La zone « Détail Élément Sélectionné » affichera le détail de l'élément sélectionné. Elle permettra à l'utilisateur de créer, modifier et supprimer cet élément.

Fonctionnalités spécifiques

Personnes

La recherche de personnes devra pouvoir être effectuée sur base des champs suivant : prénom, nom de famille, ville (city), date de naissance et l'adresse mail.

UE

La recherche des UEs devra pouvoir être effectuée sur base des champs suivant : nom et description de l'UE, sa section, nom d'une de ses capacités.

Consignes

Quelques règles à suivre dans le développement de votre code

L'application sera développée dans les langages Java/JavaScript/HTML/CSS, en utilisant à bon escient les technologies vues aux cours. Les données seront stockées dans une base de données PostgreSQL se trouvant sur le serveur Titus de l'Institut.

L'anglais sera systématiquement utilisé pour définir les noms de tous les identificateurs dans votre code (classes, fonctions, méthodes, variables, ...)

Le nom du projet Netbeans (et de votre application) sera : UE_XY, où X est la première lettre de votre prénom et Y la première lettre de votre nom.

La structure du projet définie dans ce document sera respectée.

L'utilisation de la technologie AJAX sera maximisée.

Les champs de recherches correspondants aux entités liées seront présentés sous forme de zones de textes et non sous forme de listes de choix.

Vous veillerez encore à :

- Respecter les conventions de dénomination standard de Java,
- Construire les requêtes SQL en utilisant les paramètres.

Un rapport doit être réalisé :

- Justifiant pour chaque fonctionnalité à implémenter la méthode de résolution et la technologie utilisée.
- Expliquant la structure de la base de données.
- Expliquant la structure du code source de l'application (découpe en packages, modules, classes)
- Expliquant la procédure d'installation de l'application et les prérequis logiciels permettant de la réaliser.
- Expliquant les procédures de démarrage et d'utilisation de l'application.

Échéances

Pour la partie SGBD, l'archive zip du projet Netbeans ainsi que le rapport (pdf) seront placés sur la plate-forme Moodle de l'ISL pour le mardi 6 décembre 2022.

Environnement de développement

Client

- Navigateur récent (Internet Explorer, Firefox, Chrome).
- HTML 5 – CSS 3 – JavaScript 6
- Systèmes d'exploitation : Linux et Windows

Serveur

- Application Web Java – Java SE 11 + Servlet
- PostgreSQL
- Linux

Outil de développement

Netbeans 15 (sous Linux ou Windows ou Mac)

Évaluation

L'évaluation est directement basée sur les capacités terminales des unités de formations « Projet de développement Web » et Projet de développement SGBD ».

Les capacités terminales sont divisées en deux parties : « seuil de réussite » et « degré de maîtrise ».

Pour réussir l'unité de formation, il faut avoir acquis TOUTES les capacités du seuil de réussite, 50% des points sont alors attribués (en d'autres termes vous avez 50 sur 100).

CAPACITÉS TERMINALES Projet de développement Web

Pour atteindre le seuil de réussite, l'étudiant sera capable :

au départ d'un cahier des charges proposé par le chargé de cours, face au matériel et au logiciel adéquat et face à une structure informatique opérationnelle connectée à Internet, disposant des logiciels appropriés et de la documentation nécessaire, en utilisant le vocabulaire technique et l'orthographe adéquate, et en respectant les normes et standards en vigueur,

- de créer et d'exploiter des scripts clients basés sur des classes prédéfinies ;
- de créer et d'exploiter ses propres classes ;
- de créer et d'exploiter des scripts basés sur une bibliothèque tierce ;
- d'envoyer des informations venant du client vers le serveur et de les traiter ;
- de générer un ensemble de pages web contenant un système de navigation et un contenu dynamiques intégrant formulaires et résultats.

Pour la détermination du degré de maîtrise, il sera tenu compte des critères suivants:

- les techniques de programmation utilisées,
- le degré de pertinence des commentaires dans le code,
- la lisibilité du code,
- le niveau de fiabilité des tests,
- le degré d'autonomie atteint.

CAPACITÉS TERMINALES Projet de développement SGBD

Pour atteindre le seuil de réussite, l'étudiant sera capable :

dans le respect des consignes données, en disposant d'une station informatique opérationnelle équipée d'un logiciel « Bases de données », d'un outil de développement et sur base d'un cahier des charges fourni par le chargé de cours, en utilisant les dernières techniques d'accès aux bases de données,

- d'élaborer et de défendre un dossier technique reprenant :
 - le schéma de la base de données,
 - l'expression des contraintes en langage usuel,

- la documentation du code et la gestion des erreurs ;
- d'implémenter une base de données et l'intégrité des données;
- de programmer, de tester et de défendre la programmation de l'interface visuelle qui permet la gestion des données.

Pour la détermination du degré de maîtrise, il sera tenu compte des critères suivants :

- l'utilisation pertinente des procédures développées,
- le niveau de fiabilité des tests,
- le degré d'autonomie atteint,
- l'utilisation judicieuse du vocabulaire informatique.