

Mettre en place une solution informatique pour l'entreprise - 2023-2023

Objectifs:

Créer un site web dynamique utilisant les langages HTML, CSS, PHP, Javascript, les sessions, des bases de données SQLite et la technique de programmation AJAX afin de mobiliser vos connaissances acquises lors des TD/TP des ressources R109-WEB1 & R209-WEB2 & R207-BDR.

Apprendre à organiser et à gérer les complexités des interactions entre les différents composants d'une application Web.

Rappels : Cette SAé a un coefficient 25 dans l'Unité d'Enseignement RT3

1. CAHIER DES CHARGES

Vous devez réaliser en binôme, désigné par tirage au sort, **les pages d'une application WEB** en fonction d'une base de données (avec au moins deux tables) qui vous a été attribuée (voir à la fin) + une base de données dédiée à l'authentification avec des comptes administrateurs/utilisateurs. L'application doit mettre en oeuvre tous les concepts vus dans le module complétés ou modifiés par les points suivants.

Certaines pages ou fonctionnalités seront à faire par un étudiant et d'autres par le second étudiant comme indiqué ci dessous dans le sujet. Un travail commun est à faire pour fusionner votre application et s'assurer du fonctionnement global.

1.1. HÉBERGEMENT DE L'APPLICATION

Vous pouvez utiliser un serveur WEB local pendant la phase de développement, mais vous devrez déposer avant et après chaque séance votre application sur le serveur WEB interne R&T et vous devrez mettre en place une solution PLATFORM-AS-A-SERVICE (PAAS) AVEC MICROSOFT AZURE comme vue en R202-SYS2 (cela facilitera votre travail collaboratif).

1.2. MAQUETTE DE L'APPLICATION

La maquette est libre mais vous devez disposer d'au minimum 4 zones dans vos pages (un header avec les informations de l'utilisateur connecté, un menu horizontal, une zone principale et un footer). Attention votre page doit être visuellement différente de celle étudié en mini projet (**sinon MALUS - 2pts**).

1.3. ORGANISATION DES FICHIERS

Organisez vos fichiers dans des dossiers explicites avec un encodage des fichiers en UTF-8.

1.4. PAGE DE CONNEXION CONNEXION.PHP (À FAIRE PAR LE 1^{ER} ÉTUDIANT)

Vous devrez créer une page **connexion.php** :

- elle proposera un formulaire de connexion
- elle redirigera l'utilisateur vers la page **index.php** en cas d'authentification réussie et une session utilisateur sera créée avec un statut (administrateur ou utilisateur selon une information issue de la BDD). Dans le cas contraire on restera sur la même page. Dans les deux cas, on enregistrera dans un fichier de log (inaccessible via HTTP) le login, l'adresse IP et l'heure de tentative de connexion avec le statut de connexion (connexion réussie ou non) ainsi que le statut de la personne en cas de réussite.
- **Un script Javascript vérifiera que le login saisi contient un moins une majuscule et un caractère spécial (!,*, etc.) , sinon l'envoi du formulaire sera bloqué (Option 1)**

1.5. PAGE INDEX.PHP (À FAIRE PAR LE 1^{ER} ÉTUDIANT ET LE 2^{ÈME} ÉTUDIANT)

Cette page ne sera accessible qu'en présence d'une session (administrateur ou utilisateur).

En l'absence de session, l'utilisateur sera redirigé vers la page **connexion.php**

Cette page doit afficher

- en permanence un tableau HTML avec tous les éléments pertinents disponibles dans la base de données (RQ : ce tableau ne doit pas être la table des utilisateurs) **(1^{er} étudiant)**
- un menu général qui doit être différent selon que l'utilisateur connecté soit « administrateur » ou « utilisateur » **(1^{er} étudiant)**
- un second tableau HTML dont les données auront été filtrées avec un formulaire (dynamique) de choix via la technique **AJAX (2^{ème} étudiant)**

1.6. PAGE INSERTION.PHP (À FAIRE PAR LE 2^{ÈME} ÉTUDIANT)

Cette page ne sera accessible qu'en présence d'une session « administrateur ».

- En l'absence de session, l'utilisateur sera redirigé vers la page **connexion.php**.
- En l'absence de droit, l'utilisateur sera dirigé vers la page **index.php**

Cette page doit permettre d'insérer un élément de votre choix dans la base de données, le formulaire doit contenir un formulaire dynamique.

L'insertion ne doit pas se faire sur la base « utilisateurs » mais au moins sur une table de votre BDD fournie.

La vérification du formulaire doit se faire au niveau du client en Javascript (Option 2).

La vérification du formulaire (insertion) doit se faire au niveau du serveur avec une CAPTCHA (Option 3).

1.7. PAGE MODIFICATION.PHP (À FAIRE PAR LE 1^{ER} ÉTUDIANT)

Cette page ne sera accessible qu'en présence d'une session « administrateur ».

En l'absence de session, l'utilisateur sera redirigé vers la page **connexion.php**.

En l'absence de droit, l'utilisateur sera dirigé vers la page **index.php**

Cette page doit permettre de modifier un élément de la base de données.

- Le choix de l'élément à modifier se fera à l'aide d'un formulaire dynamique. **La vérification du formulaire (modification) doit se faire au niveau du serveur avec une CAPTCHA (Option 4).**
- En réponse à ce choix vous afficherez un formulaire pré-rempli avec les données de l'élément à modifier. L'envoi de ce formulaire (pré-rempli) pour valider les modifications se fera soit
 - **en mettant en œuvre la technique de programmation AJAX. Le serveur retournera (via AJAX) un affichage HTML des informations mises à jour (Option 5).**
 - **en rappelant la même page avec l'affichage des informations mises à jour (Option 6).**

1.8. PAGE SUPPRESSION.PHP (À FAIRE PAR LE 2^{ÈME} ÉTUDIANT)

Cette page ne sera accessible qu'en présence d'une session « administrateur ».

En l'absence de session, l'utilisateur sera redirigé vers la page **connexion.php**.

En l'absence de droit, l'utilisateur sera dirigé vers la page **index.php**

Cette page doit permettre de supprimer un élément de la base de données.

- Le choix de l'élément à modifier se fera à l'aide d'un formulaire dynamique. **La vérification du formulaire (modification) doit se faire au niveau du serveur avec une CAPTCHA (Option 4).**
- En réponse à ce choix vous afficherez le tableau HTML avec tous les éléments pertinents disponibles dans la base de données

1.9. BONUS : LE STYLE DU SITE SERA DÉFINI À L'AIDE D'UN FRAMEWORK

Il s'agit d'un bonus et ne doit pas être la priorité de votre application.

Vous pouvez utiliser le framework déjà vus en WEB1 (Bootstrap)

2. LES SUJETS

2.1. LES BDD

- Certaines bases de données qui vous sont proposées sont composées de 2 tables, avec une relation entre ces 2 tables. C'est le schéma minimum de base.
- D'autres bases de données fournies sont plus complexes, dans ce cas vous n'êtes pas obligés de mettre en œuvre les insertions, modifications et suppressions sur toutes les tables (à discuter avec votre enseignant de TP au cas par cas).
- Vous devez ajouter au moins 10 enregistrements par table (directement à partir de votre gestionnaire de BDD, DB Browser par exemple) afin d'avoir du contenu à gérer (sinon MALUS - 2pts).
- Vous pouvez, pour le besoin de votre application, améliorer votre base de données en ajoutant des colonnes, en modifiant le nom d'un champ, en modifiant la longueur d'un champ... tant que le minimum reste.
- Vous devrez ajouter un champ à une des tables de votre BDD (au choix) qui contiendra le chemin d'une image à faire afficher par votre application WEB (Par exemple l'image des livres contenu dans votre BDD si votre site gère une bibliothèque)
- Lors de l'ajout d'un enregistrement, vous devrez veiller à l'intégrité de votre base.
- Pour la gestion des utilisateurs (au moins un admin et un utilisateur simple), vous devrez créer une BDD et y ajouter cette table.

```
CREATE TABLE "comptes" (  
  "login" VARCHAR PRIMARY KEY NOT NULL,  
  "motdepasse" VARCHAR DEFAULT lannion,  
  "statut" VARCHAR NOT NULL DEFAULT utilisateur)
```

Remarque: l'insertion demandée ne doit pas se faire sur la BDD qui contient la table « comptes ».

2.2. LES OPTIONS IMPOSÉES

On vous impose le traitement de 3 options, mais vous pouvez faire les autres en bonus si vous le souhaitez dès lors que les options imposées ont été réalisées et réussies !

3. MODALITÉS

3.1. GROUPES

Le travail se fera en binôme désigné aléatoirement.

Groupe A			Groupe B			Groupe C		
Sujet	Options	Binôme	Sujet	Options	Binôme	Sujet	Options	Binôme
Matériel Fournisseur	1-3-5	CHICHEPORTICHE REVERDY	ClientVentes	1-3-5	GRANJEON KEROULAS	Représentants	1-3-5	MOOTOOSAMY GUYON
avisClientsProduits	2-4-6	JOUENNE VOLLE	CaveABieres	2-4-6	BEAUPEL GORE	Utilisateurs Paniers Produits	2-4-5	THÉPAUT BUCAILLE
Lycée	1-3-5	BOEZ BRUNET	RTAbsences	2-3-5	MARCO BEAUDOUIN	BDTheque	1-3-5	KLAPSIA HOCHARD
Ludotheque	2-4-6	DOUDET SANDALI	PilotesMoto	2-4-6	FROGER WEIS	Brocantes	2-4-6	COUQUEBERG YOUSFI
BieresElBrasseurs	1-2-5	COINTO KERDRAON	CaveAVin	1-3-6	HEYRENDT CRON-DUPON	ProduitsGourmands	1-3-6	LEBLANC-JEZEQUEL CARLOTTI
Bibliotheque	2-4-6	LEMICHELET EZZITOUNI	Contacts	2-4-5	GRANNEC GOUIN	Poemes	2-4-6	PIERROT JOUET
GestionNotes	2-3-5	OUATTARA TRIHINE						
AnnouncesAutosPlus	2-3-5	THIERY GOUEZ	Cave	2-3-5	CUELLAR BATIL	TeamPlayerPays	2-3-5	LE FEVRE DE SOUSA CASTRO
Equipe Recherche	2-4-5	HERVE PERISSAT	MatièresNotes	2-4-5	PENCRANE LOSSER	Avions	2-4-5	BODILIS MAHE M
produitsLivraison	1-4-6	MARTIN VIVIER	Menu Restaurant	1-4-6	GRASSET ROINET	CDTheque2	1-4-6	QUEFFELEC COUTEILLE
JeuxVideos	1-3-6	TOURNEUX VILFAILLLOT	EDT	1-3-5	SORG WEINMA SEVESTRE	Quizz	1-3-5	DUCCLOS NICOLAS
LocationMatériel	1-4-5	JACO REHAULT	EMPLOYES	1-4-6	SALOU OSKAN	Etudiants_Groupes	1-4-6	DIMASCIO MAHE H
Fleurs	2-3-6	PAUL CASANUEVA PERIANEZ						
Options	1 ou 2	3 ou 4	5 ou 6					
1	Un script Javascript vérifiera que le login saisi contient un moins une majuscule et un caractère spécial (+, -, etc.) , sinon l'envoi du formulaire sera bloqué.							
2	La vérification du formulaire doit se faire au niveau du client en Javascript.							
3	La vérification du formulaire doit se faire au niveau du serveur avec une CAPTCHA.							
4	La vérification du formulaire doit se faire au niveau du serveur avec une CAPTCHA.							
5	soit en mettant en œuvre la technique de programmation AJAX. Le serveur retournera (via AJAX) un affichage HTML des informations mises à jour							
6	soit rappelant la même page avec l'affichage des informations mises à jour							

3.2. PLANNING

date			20/3	27/03/23	3/4	10/4	17/4	24/4	1/5	8/5	15/5	22/5	29/5	5/6	12/6	19/6
semaine	calendaire		12	13	14	15	16	17	18	19	20	21	22	23	24	25
C/TD 9H TP 15H	CM	3	2	0		1										
Initiation au développement WEB			CM1: Intro	CM2: Form HTML5 + GET/POST PHP		CM3: coté serveur + Sessions										
	TD	10,5		1,75	3,5	1,75			1,75	1,75						
			TD1: PHP	TD2: PHP	TD3: PHP GET POST accès Bdd	TD4: PHP Sessions			TD5: PHP une application	TD6: PHP une application (Suite)						
	TP	8			1,75	1,75			1,75	1,75						
				TP1: PHP		TP2: PHP GET POST accès Bdd			TP3: PHP Sessions	TP4: PHP une application (Suite)						1 Test TP
SAE23		16														
C/TD 3H TP 14H PRJ12	CM	2							2	0	0					
SAE23 Mettre en place une solution informatique pour l'entreprise										CM3: coté client Javascript	Projet					
Encadré Info	TD	9									3,75	1,75	1,75	1,75		
											TD7 Javascript + TD8 AJAX	Projet	Projet	Projet		
	TP	5								2	2	A REVOIR!!!				
										TP5: PHP une application (Fin)	TP Javascript / Ajax					1 Test TP
BDD (à part)	TP	0														
Heures Projet 12H Non Encadrée	Non	9								1	2	2	1,75	1,75		0,5 Soutenance
										CM4: Côté client Javascript +	Projet	Projet	Projet	Projet		

3.3. SUIVI DU TRAVAIL (À CHAQUE SÉANCE)

Vous devrez déposer après chaque séance dans votre espace de travail collaboratif (cf Mattermost) :

- Point d'avancement séance n°XX
- Lieu de travail : Salle XX
- Méthode de travail : binôme/seul
- Tâches prévues pendant la séance :
- Tâches traitées pendant la séance :
- Problèmes rencontrés :
- Tâches prévues avant la prochaine séance :

Vous poserez vos questions exclusivement sous Mattermost (pas de mail !)

Une mise à jour de votre site devra être accessible depuis le serveur WEB interne du département R&T avant et après le début de chaque séance. Vous devez aussi utiliser un hébergement personnel (ex Azure), et vous devrez communiquer l'URL sous **Mattermost** (<https://sae-web-iut-rt-lannion.cloud.mattermost.com/sae23-but-rt-lannion>) dans votre canal privé.

Le suivi se fera en groupe « TD ».

Vous devez démarrer votre projet avant la première séance encadrée (cf planning).

3.4. ÉVALUATION (CF GRILLE FOURNIE ULTÉRIEUREMENT)

Les critères pris en compte pour la notation sont :

- Le respect du cahier des charges minimum,
- La conformité HTML5, CSS
- La structuration/qualité/clarté du code commenté,
- L'ergonomie du site de l'application
- Le respect du format du dépôt et des délais de dépôt

3.4.1. SOUTENANCE

Une soutenance de 10' maximum aura lieu en semaine 25.

Cette soutenance sera en fait une démonstration de votre application en respectant l'ordre de la grille de validation. **Votre site devra être accessible lors de la soutenance depuis le serveur WEB interne du département R&T**

3.4.2. TRAVAIL À RENDRE

Il faudra déposer sur l'ENT le **jour** de votre soutenance :

- Plusieurs vidéos qui capturent votre écran (ex logiciel OBS Studio) et vos commentaires oraux en exposant **tous les tests** de votre application avec :
 - les logins et mot de passe de l'administrateur du site et d'un utilisateur,
 - votre BDD avec tous les enregistrements,
 - les accès aux pages sans être connecté,
 - les accès aux pages en étant connecté en tant que simple utilisateur,
 - les accès aux pages en étant connecté en tant qu'administrateur,
 - le test d'affichage conditionnel avec la liste déroulante,
 - le test d'ajout,
 - le test de modification,
 - la conformité HTML de toutes vos pages,
 - la conformité CSS.

Vous complèterez cette vidéo en montrant l'organisation de vos fichiers et dossiers et vous ouvrirez tous les fichiers que vous avez créés en les parcourant (scroll intégral) afin de les capturer sur la vidéo et de les commenter rapidement. La réalisation de cette vidéo vous permettra aussi d'être au point pour la soutenance. **Chaque étudiant doit présenter sa partie.**

- les codes sources commentés du projet contenu dans une archive au format **zip** sur l'ENT nommée **NOM1_NOM2_SAE23_WEB2.zip**.