

Projet CIR2

GESTION DE DONNÉES D'OBJETS CONNECTES

Evaluation


Mis à jour le 22/05/2017

michael.aron@isen-ouest.yncrea.fr
yann.le-ru@isen-ouest.yncrea.fr
didier.le-foll@isen-ouest.yncrea.fr
fabienne.provost@isen-ouest.yncrea.fr



Les deadlines...

Livrables	Limite
MCD + fichier SQL	30/05 17:30
Sources du projet	01/06 08:00



Les livrables seront à poster sur l'intranet
Tout retard sera sanctionnée (l'heure du réseau fait foi)
Les fichiers au mauvais format ou mauvais nommage seront pénalisés



Livrable BDD

- Une archive à déposer sur l'ENT (infos sur la norme à respecter sur le dépôt) contenant :
 - Fichier SQL contenant la structure de la BDD, ainsi qu'un jeu de données permettant d'effectuer les tests.





Livrable sources

- Une archive à déposer sur l'ENT (infos sur la norme à respecter sur le dépôt) contenant :
 - Fichier README indiquant les étapes à effectuées pour que le site fonctionne sur n'importe quel serveur (les informations suivantes doivent être facilement paramétrables : prérequis, mise en place de la BDD, configuration accès BDD, URL du site, configuration apache éventuellement, ...)
 - Code source :
 - PHP, HTML, Javascript, programme C++, scripts shell, ressources (images, ...), serveur temps réel si besoin, ...





Informations

- Les barèmes indiqués dans ce document sont donnés à titre indicatif
- Le but de ce document est d'indiquer les fonctionnalités attendues et de donner un ordre de priorité dans la réalisation des tâches



Barème prévisionnel général

Partie	Sous-partie	Barème
Réunion client	Environnement de dev/prod	0,5
	Architecture	0,5
	MCD	2
	Maquette Web	1
	Présentation C++	1
Web	Fonctionnalités (recette)	6
C++	Fonctionnalités (recette)	6
Alertes	Fonctionnalités (recette)	3
	Total	20



Informations Web

- Se concentrer sur la partie cliente !!!
- Répartition probable :
 - Client : 100%
 - Admin : bonus...



Barème prévisionnel web : client

#	Intitulé		Barème
1	Structure générale	MVC	0
		Ergonomie	
		Navigation	
		Mobile	
		Configuration Appli (BDD, ...)	
2	Gestion utilisateurs	Inscription au service	
		Procédure récupération mot de passe	
		Modification du profil	
		Ajout/Modification/Suppression de superviseurs	
	Total		



Barème prévisionnel web : client

#	Intitulé		Barème
3	Affichage données	Sélection données à afficher	
		Mode temps réel	
		Mode historique : sélection d'une plage temporelle	
		Courbes en mode temps réel	
		Courbes en mode historique	
		Affichage valeurs numériques pour des points d'intérêts choisis par l'utilisateur	
		Seuils d'alertes (humidité, température, CO2)	
		Sélection des capteurs à afficher (ergonomie)	
	Total		

Barème prévisionnel web : client

#	Intitulé		Barème
5	Accès application	Utilisateur	
		Superviseur	
		Pas d'accès pour d'autres utilisateurs	
	Total		



Barème prévisionnel web : admin

#	Intitulé		Barème
1	Utilisateurs	Ajout	
		Modification	
		Suppression	
		Association des superviseurs	
	Total		



Barème prévisionnel web : admin

#	Intitulé		Barème
2	Chambres	Ajout	
		Modification	
		Suppression	
		Association avec un utilisateur	
	Total		





Barème prévisionnel C++

#	Intitulé		Barème
1	Fonctionnalités	Accès BDD paramétrable	
		Communication : websocket : récupération des données IP/port	
		Analyse des données Json	
		Insertion des données capteurs dans la BDD	
		Récupération de données provenant de plusieurs IP	
		Connexion à la BDD pour la lecture des IP/ports de connexion	
	Total		





Barème prévisionnel C++

#	Intitulé		Barème
2	Code	Séparation code BDD/calcul	
		Commentaires compatibles DOXYGEN	
		Compilation code source	
		...	
	Total		





Alertes

#	Intitulé		Barème
1	Détection en temps réel	Traitement lors de l'arrivée de données	
		Alerte seuil	
		Alerte critères complexes (ex : sur la durée...)	
2	Communication	Envoi d'un mail	
		Affichage de l'alerte dans la partie Web	
	Total		



VALIDATION DES FONCTIONNALITÉS



Processus

- Séparation de la validation des différentes parties
 - Interface Web cliente
 - Entrées : informations venant de la BDD (possibilité de peupler la BDD manuellement)
 - Sorties : aucune
 - Programme C++
 - Entrées : paramètres, fichiers (simulation des données capteurs), json via sokets...
 - Sorties : BDD
 - Alertes (pas forcément indépendant...)
 - Entrées : informations venant de la BDD, fichiers(simulation des données capteurs), json via sokets
 - Sorties : affichage console, affichage fichier, envoi mail, ...





BDD

- Un fichier SQL contenant :
 - La structure de la BDD
 - Des données permettant de tester les fonctionnalités de l'application Web cliente : utilisateurs, chambres, données capteurs, ...





Mise en garde !

- Les informations précédentes sont à lire **attentivement**
- Une fonctionnalité ne pouvant pas être testée aisément ne sera pas validée (pas d'opérations manuelles à effectuer lors de la recette – ajout dans la BDD, modifications dans le code source...)



PROCESSUS D'ÉVALUATION



Préparation recette (BDD)

- **Etape 1** : Les **fichiers sources** doivent être postées sur l'intranet **avant l'heure indiquée** sous format compressé (*.zip ou *.rar)
- **Etape 2** : L'enseignant importe le fichier SQL sur un serveur de test (attention, le nom de la BDD ainsi que l'utilisateur sont choisis par l'enseignant → paramétrables)
 - Si erreur lors de l'import : repassage (3 points en moins)





Recette Web

- **Étape 1** : L'enseignant **télécharge** l'archive **sur le serveur de test** dans un **répertoire de test choisi par l'enseignant**
- **Étape 2** : L'enseignant suit les instructions décrites dans le fichier README pour installer et configurer votre application
 - Si autres actions à réaliser pour que le site soit opérationnel : repassage (3 points en moins)



Recette Web

- **Étape 3:** L'enseignant vérifie avec les étudiants les différents points de la recette.
 - Si impossibilité de faire les tests : validation sur votre serveur de production (1 points en moins)
 - Si impossibilité de faire les tests : repassage (3 points en moins)



Recette Web

- **Étape 4** : La qualité du code est analysé par un outil de qualimétrie logicielle ([SONAR](#))
 - Intégrité,
 - Modularité
 - Nombres de variables locales et globales
 - Commentaires, ...
- **Étape 5 (à posteriori)** : Les codes de chacun des binômes sont comparés entre eux avec un outil anti-plagiat ([Baldr](#))



Recette C++

- **Étape 1** : L'enseignant télécharge l'archive sur le serveur de test dans un répertoire de test.
- **Étape 2** : L'enseignant suit les instructions décrites dans le fichier README pour installer et configurer votre application
 - Si autres actions à réaliser pour que le logiciel soit opérationnel : repassage (3 points en moins)
- **Étape 3** : Les étudiants et l'enseignant compilent ensemble le projet
 - Si erreur de compilation : repassage (3 points en moins)



Recette C++

- **Étape 4** : L'enseignant vérifie avec les étudiants les différents points de la recette
 - Si impossibilité de faire les tests : validation sur votre serveur de production (1 points en moins)
 - Si impossibilité de faire les tests : repassage (3 points en moins)



Recette C++

- **Étape 5** : La qualité du code est analysé par un outil de qualimétrie logicielle ([SONAR](#))
 - Intégrité,
 - Modularité
 - Nombres de variables locales et globales
 - Commentaires, ...
- **Étape 6 (à posteriori)** : Les codes de chacun des binômes sont comparés entre eux avec un outil anti-plagiat ([Baldr](#))



FIN

Des questions?

