

Examen 5IDV2 - Juin 2020

Chargé de cours : Benjamin Delbar
IFOSUP WAVRE

1. Matière vue
2. Projet à réaliser
3. Examen oral
4. Répartition des points
5. FAQ

1. Matière vue

Vous trouverez ci-dessous une liste récapitulative de la matière vue en classe.

Cette matière est donc sujette à être implémentée dans votre projet.

Création d'une base de données mySQL par l'interface phpMyAdmin

Création de triggers SQL par l'interface phpMyAdmin ou en SQL directement

Connexion à une base de données par l'intermédiaire du PDO en PHP

Théorie sur la protection des données sensibles en vue de stockage (hash, salt, pepper)

Injections SQL et comment s'en protéger

Injections de scripts et comment s'en protéger

Utiliser du javascript et la librairie jQuery pour améliorer l'expérience utilisateur

Requête XHR : principe et cas d'utilisations avec la librairie jQuery

Séparer la logique de programmation entre les objets (entities) et l'accès aux données (DAL)

Création de Data Access Object (DAO)

Interface de gestion CRUD (Create-Read-Update-Delete) d'un modèle de données - avec et sans XHR

Utiliser des principes de POO pour créer une classe abstraite DAO générale

Créer un routeur et procéder à la réécriture d'urls

Créer une architecture "MVC" pour séparer la logique du traitement des données de la logique d'affichage des données

Créer un système de template pour générer les vues

Design Pattern Strategy : Comment régler un problème courant lié à l'héritage

Principe de persistance des données : Soft Delete & Hard Delete

Principe d'instanciation au runtime / hard-codé

Utilisation des developer tools pour analyser les requêtes

Différents type de communication client - serveur : requête classique, polling, long-polling, push (via les server sent events)

Lecture et écriture dans des fichiers textes

Communiquer avec des API externes via des endpoints (routes) Et traiter le résultat reçu en JSON (cat facts, pokeAPI)

Lire et utiliser la documentation d'une API pour l'intégrer correctement

Différents moyens de faire persister une authentification (Session ID & Token)

Forces et faiblesses des session id et des tokens

Attaques CSRF : Qu'est-ce que c'est et comment s'en protéger

Mot de passe : théorie et implémentation des fonctions PHP password_hash et password_verify

Persistance d'une authentification grâce aux cookies

Indexé et rendre unique un champ en base de données

Moteur de stockage - MyISAM & InnoDB

Clef étrangère et contrainte "restrict - cascade"

Implémenter des rôles différents aux utilisateurs

Améliorer ses DAO pour gérer les relations

2. Projet à réaliser

En vue de préparer votre examen, il vous est demandé de réaliser un projet complet. De l'analyse de la demande client jusqu'à l'implémentation du code en passant par la création d'une base de données et la rédaction d'un rapport.

Vous avez le choix parmi **3 propositions de projets** (voir en annexe). Les 3 propositions comprennent des contraintes assez similaires. Il n'y a donc pas de choix stratégique avec un projet plus "facile" qu'un autre.

Je vous conseille donc de prendre le projet qui vous parle le plus.

Peu importe le projet choisi, il faut **respecter les contraintes communes** présentes dans le point ci-dessous.

Les fiches de projets comportent un texte reprenant les désirs du client (reprenant des informations pour vous aider à réaliser les fonctionnalités obligatoires et complémentaires), une liste des fonctionnalités obligatoires et une liste des fonctionnalités complémentaires.

Le projet comporte des **fonctionnalités obligatoires** et des **fonctionnalités complémentaires**, les fonctionnalités obligatoires sont, vous l'aurez deviné obligatoire, et les complémentaires vous permettront d'augmenter votre degré de maîtrise, il est vivement conseillé d'essayer d'intégrer l'une ou l'autre fonctionnalité complémentaire évidemment. Libre à vous d'ajouter d'autres fonctionnalités qui vous paraissent intéressantes et qui ne sont pas reprises, notez les bien dans votre rapport.

Les 3 derniers cours sont dédiés à vous laisser la possibilité de réaliser le projet "en classe". Je serai présent et disponible à répondre à vos questions directement pendant le cours.

Comme d'habitude, vous pouvez toujours me poser vos questions en dehors des heures de cours via un message privé sur Discord ou par mail.

Si nécessaire et à la demande de plusieurs élèves, il est possible de prendre une partie du cours pour revoir un point en particulier. Par exemple : "Revoir comment fonctionne le polling".

Votre présence sur discord dans le canal "Cours en session" est requise pendant ces 3 cours, **les présences seront prises au début du cours** et libre à vous de rester par la suite ou non.

Contraintes communes

Le projet doit fonctionner sur un serveur Apache, avec une base de données **mysql** et être écrit en **PHP** (version ≥ 7.0). Un WAMP ou MAMP à jour devrait vous fournir cela. Si vous avez des doutes, il est toujours possible de m'envoyer votre travail à l'avance en me demandant de vérifier la compatibilité.

Il n'est pas autorisé d'utiliser un framework, et est autorisée uniquement la librairie **jQuery** ainsi que des fichiers de style pour faciliter la réalisation d'une interface.

Au niveau du code :

Vous devez implémenter le **design pattern Strategy** pour **au moins 1 autre fonctionnalité** que le choix du delete vu en classe.

Vous devez utiliser une architecture comprenant des **modèles, des contrôleurs et des vues**.

Vous devez **sécuriser vos formulaires** et le traitement des données contre les attaques de type CSRF, les injections SQL et les injections de scripts.

Vous devez **protéger les données sensibles** (password) avec une sécurité "Hash + Salt" au minimum.

Certains modèles (au choix) doivent permettre ou obliger d'être supprimer de manière soft.

Vous devez utiliser des **requêtes XHR** pour **au moins 1 fonctionnalité**.

Il n'est pas obligatoire d'utiliser un routeur mais c'est fortement conseillé.

Il n'est pas obligatoire d'utiliser du javascript (à l'exception de la requête XHR) mais cela est fortement conseillé.

Un minimum d'effort doit être apportée à la mise en page : Pas besoin de CSS, mais une structure HTML correcte, ajouter des labels à vos champs, ajouter un titre à vos formulaires.

Vous devez **faire persister l'authentification** d'un utilisateur pendant, au minimum, la durée de la session.

Faites attention aux erreurs SQL, transmettez un message compréhensible pour un non-développeur lorsqu'une erreur arrive.

Faites attention aux URL, est-ce que votre projet fonctionne toujours si vous le déplacez dans un autre dossier, dans un sous dossier?

Délivrables

Code source

Rendu sous forme d'archive .zip ou .rar, ou sous forme de lien vers un repository git publique. Le code source **doit contenir un dossier dump** reprenant le dump de votre base de données.

Il n'est pas obligatoire que votre code soit commenté mais il vous est demandé de nommer vos fonctions, méthodes, classes et variables le plus clairement possible.

Base de données

Le dump de la base de données doit être pré-rempli de quelques entrées pour chaque

tables.

Rapport

Le rapport doit être rendu au format PDF, et être envoyé en même temps que le code source.

Le rapport doit contenir

- Une page de garde avec le nom de votre projet, ainsi que votre nom et prénom.
- Le schéma relationnel de votre base de données
- La liste des routes de votre projet sous le format standard précisé en annexe
- Une description des éventuels problèmes rencontrés et des solutions trouvées
- La liste des fonctionnalités implémentées
- *Optionnel* : Une zone libre où vous pouvez mettre en avant une fonctionnalité / un bout de code / un diagramme / ...
- Une autocritique de quelques lignes par rapport à votre travail

Echéance

Le travail est à rendre (Code source + Dump Base de données + Rapport) pour **le dimanche 7 juin 2020 à 23:59 au plus tard.**

Envoi après le 7 juin : 10% de pénalité par jour de retard entamé

Envoi après le 10 juin : seconde session

Ce qui est important

Une fonctionnalité développée à 100% et testée vaut mieux que 2 fonctionnalités qui ne marchent qu'à moitié.

L'important est de montrer par votre code et votre rapport que vous avez compris ce qui a été vu en classe et que **vous comprenez ce que vous avez codé.**

Il est préférable de faire un travail "moche" mais dont vous comprenez chaque ligne, plutôt qu'un travail "beau" qui est un frankenstein de morceaux de codes que vous ne comprenez pas.

Liste des projets

Vous trouverez ci-dessous une description des 3 projets possibles. Une liste plus détaillée avec les fonctionnalités est disponible dans les documents correspondants.

Projet A : Online Class

Suite à la crise du Coronavirus, vous êtes contactés par l'IFOSUP pour créer une plateforme de cours en ligne.

Projet B : E-Score

Un site d'information sur l'esport à décider de mettre en place une plateforme pour les équipes ou elles peuvent s'inscrire et ajouter leurs résultats.

Projet C : My Building

Un syndic vous a contacté pour créer une plateforme permettant de gérer les problèmes qui surviennent dans les immeubles sous leur responsabilité.

3. Examen oral

Inscription à l'examen oral

Il faut impérativement s'inscrire sur le Google Forms.

Le lien du google form : <https://forms.gle/s8wqzLu4uLQWZ8eb6>

La date limite d'inscription est le **Jeudi 4 juin 2020 à 23:59**.

Les horaires de passages seront disponible sur Discord au plus tard le Samedi 6 juin.

Vous pouvez échanger votre heure de passage avec quelqu'un d'autres, dans ce cas merci de me prévenir.

Déroulement de l'examen oral

L'examen oral dure entre 15 et 30 minutes (grand maximum) et se déroule en étapes :

10 minutes maximum : Présentation de votre projet, de ce que vous avez implémenter, des choix techniques que vous avez fait. Vous pouvez ici mettre en avant une fonctionnalité ou un bout de code dont vous êtes particulièrement fier ou qui est intéressant par exemple.

15 minutes maximum : Session de questions & réponses.

Exemples de question :

- *Justifier pourquoi vous avez choisi d'implémenter un Design Pattern Strategy pour gérer les comportements de suppression.*

- *Pourquoi avoir choisi d'utiliser un trigger SQL pour gérer la mise à jour du stock des produits?*

- *Dans ton UserDAO, ta méthode save() override la méthode save() de ton DAO abstrait, pourquoi?*

Les questions sont individuelles et faites en fonction de votre projet. Néanmoins certaines questions plus génériques seront tirées aux hasards parmi une liste, tel que :

- *Si le client te demande de permettre à un produit d'avoir plusieurs catégories, qu'est-ce que cela implique comme changement dans ton code?*

- *Comment t'y prendrais-tu pour ajouter la fonctionnalité : "Une chatroom est disponible sur chaque fiches produits"*

5 minutes : Feedback sur le cours & formalités administratives

L'examen oral se déroule sur Discord, avec pour vous la possibilité de partager votre écran pour plus facilement répondre aux questions / montrer une présentation / ...

L'examen oral est enregistré. Ceci servant de preuve de votre passage et pour vous comme pour moi, en cas de conflit potentiel, de trace de votre examen.

Il est fortement recommandé (mais pas obligatoire) de préparer un support pour votre présentation. Sous la forme d'un powerpoint ou autres.
Cela vous aidera à bien présenter votre projet et vous aidera aussi à anticiper les questions les plus probables.

Si vous ne disposez pas du matériel nécessaire pour l'examen oral, vous pouvez soit installer Discord sur votre téléphone, soit faire une demande au secrétariat pour présenter votre examen depuis l'IFOSUP.

Présence à l'examen oral

Il vous est demandé d'être présent sur discord au minimum **5 minutes avant** votre heure de passage. Profitez-en pour tester votre micro et déjà ouvrir les fichiers que vous voudriez montrer.

De mon côté

Si un retard au niveau des passages venait à s'accumuler, un message sera publié dans le channel du cours sur discord pour vous prévenir et permettre aux suivants d'arriver plus tard.

De votre côté

Il est important de me prévenir au plus tôt **via un message privé sur Discord** (et non par mail car je serai fort probablement occupé en session avec un autre élève) en cas de retard.
Dans la mesure du possible nous définirons une nouvelle heure de passage le même jour.

Un retard non prévenu, ou une absence de votre part lors de votre passage entraîne automatiquement une seconde session.

4. Répartition des points

Travail : 50%

Oral : 50%

Pour rappel, la fiche pédagogique du cours indique que l'étudiant doit être capable (pour atteindre le seuil de réussite) :

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.

Et le degré de maîtrise est déterminé par

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

Je vous souhaite un excellent travail, vous avez toutes les clefs nécessaires pour réussir largement l'examen même avec la situation particulière que nous vivons, vous êtes restés motivés et concentrés! Et je vous en remercie.

5. FAQ

J'ai trouvé un gros bug dans mon code après vous l'avoir remis : Qu'est-ce que je fais?

Cela arrive souvent, pas de stress, fixer le, et notez bien comment vous l'avez réglé et pourquoi ce bug était présent. N'hésitez pas à en parler pendant votre présentation. Ca vous permettra peut-être de récupérer une partie des points perdus.

Suite à des problèmes externes à ma volonté je me retrouve dans l'incapacité de terminer le travail au niveau attendu.

C'est des choses qui arrive évidemment. Venez m'en parler le plus tôt possible pour que nous puissions essayer de trouver une solution. N'oubliez pas qu'il y a aussi une deuxième session s'il n'est vraiment pas possible pour vous de rendre le travail ou de présenter l'oral.

Je suis bloqué sur une des fonctionnalités et je n'arrive pas à m'en sortir.

Premièrement : aller revoir le cours en question, si cela n'aide toujours pas vous pouvez venir me demander de l'aide lors des cours ou en dehors. N'oubliez pas de faire aussi appel à vos camarades de classe, la solidarité et l'entraide sont d'autant plus important avec les conditions actuelles.

Je ne me sens pas à l'aise avec l'utilisation de jQuery / Javascript, est-ce que je suis obligé d'en utiliser?

Il est demandé dans le travail d'utiliser pour au moins un cas des requêtes XHR, pour cela vous devrez donc utiliser du javascript avec (ou sans si vous préférez) l'aide de la librairie jQuery.

Pour le reste du travail, aucune obligation. Maintenant, il a été montré en classe à quel point un peu de gestion d'événements en javascript peut augmenter l'expérience d'utilisation de votre travail (et cela fait partie des compétences requises). Si vous décidez donc de ne pas utiliser de javascript, il vous faudra essayer de rendre les changements de pages (envoi de formulaire / récupération de données) les plus intuitifs et les moins gênant possible.

Annexes

Format pour la documentation des routes

La documentation d'une route doit contenir

- L'url correspondante
- La méthode utilisée (post, get)
- Les paramètres envoyés
- Les réponses émises par le serveur
- Autres informations jugées utiles si nécessaire (authentification nécessaire?)

Prenons comme exemple une route `/users/delete` qui en post reçoit en paramètre l'id du user à supprimer et renvoie en réponse un objet de type json contenant `{response: "Utilisateur supprimé"}`.

POST `/users/delete`

Paramètre	Type	Status	Description
id	int	Obligatoire	Id correspondant à l'utilisateur à supprimer

Réponse	Type	Status	Description
response: "Utilisateur supprimé"	JSON	200 - OK (done)	Réponse reçue si l'utilisateur est supprimé correctement
response: "Erreur : id non trouvée"	JSON	404 - Not found (fail)	Réponse reçue si l'id ne correspond à aucun enregistrement / l'id n'est pas précisé

Prenons comme autre exemple : une route `/products` , ne prenant pas de paramètres et qui renvoie la liste des produits.

GET `/products`

Réponse	Type	Status	Description
Liste des produits	Text/HTML	200 - OK	Renvoie la page complète avec la liste des produits