

# [IMAC S2] Programmation Back-end

## Projet Final

### Brief

---

## Sujet

Vous réaliserez un **service Web Restful** avec comme technologies :

- **PHP** en langage back (frameworks interdits)
- **MySQL** en tant que SGBD
- Côté front: **peu importe**. Tant que le client arrive à lancer des requêtes HTTP.
- **Git** pour le système de versionning (suivez les tutos !)

Le site doit être accessible en ligne (si nous n'avez pas moyen d'avoir votre propre hébergement, sachez que chaque étudiant.e de l'upem peut avoir son site perso, plus d'info : <https://cri.u-pem.fr/service-aux-etudiants/web>)

**L'API côté back est la priorité du projet.** Le but étant de pouvoir manipuler une base de données avec les 4 principales actions : **Create, Read, Update, Delete**.

Cependant **une interface côté client est demandée**. Le site doit pouvoir réaliser au moins **10 actions** sur votre base de données (le site doit avoir au moins 10 routes/endpoints). Ce service Web peut être responsive et fonctionnel sur un maximum de navigateurs Internet possible (le design peut jouer dans la note, mais c'est minime).

La base de données ne doit pas être trop grosse (**genre 5-10 tables maxi**, c'est déjà bien). **L'utilisation de Merise** pour schématiser votre bdd **est recommandée**. Vous pouvez vous servir de votre propre méthode pour modéliser votre bdd, dans ce cas justifiez-le au maximum.

**Ayez en tête que les sessions et les systèmes de login/mdp ne sont pas demandés (pas le temps de voir en cours).** Pensez alors à faire votre site qui ne soit pas restreint par l'authentification (cependant vous pouvez en intégrer dans votre site, vous faites ce que vous voulez).

**Le thème du site est libre !** Mais si vous n'avez pas d'idées, j'ai quelques exemples pour vous :

- un site marchand (gestion de commandes),
- un système de playlist,

- un dashboard,
- un pokédex-like,
- un mini réseau-social,
- un site qui note les bars du jeudimac !
- ...

## Livraison

- **Repo Git** : le ou la chef.fe de projet devra envoyer un lien du repo Git à communiquer dès le début du projet. **Date limite : samedi 16 mars à MINUIT.**
- **Pitch du projet** : À envoyer en même temps que le repo Git. Courte description de votre projet qui va RÉVOLUTIONNER le ouèbe 2.0. **Date limite : samedi 16 mars à MINUIT.**
- **Base de données** : ébauche du MCD à **envoyer avant le avant le 06 avril à MINUIT.** La version finale (ainsi toutes les autres étapes de Merise) doit y figurer **dans le rapport. Si vous ne faites pas de MCD justifiez-le au maximum.**
- **Site** : premier hébergement avec PHP qui tourne à **envoyer avant le avant le 13 avril à MINUIT** (osef si ça ne ressemble à rien). **Livraison FINALE une semaine avant la soutenance (date à préciser).**
- **Rapport** : document de 10-20 pages en format PDF. **Date limite : en même temps que la livraison finale du site (date à préciser).**
- **Soutenance** : **date à préciser (fin mai).**
- À envoyer par mail à l'adresse suivante : [hp.pascale.ho@gmail.com](mailto:hp.pascale.ho@gmail.com)
- **TOUTE LIVRAISON QUI A ÉTÉ FAITE APRÈS LA DATE LIMITE DEMANDÉE**  
**DONNERA AU GROUPE UNE PÉNALITÉ, JE SERAI INTRANSIGEANTE 😊**

## Notation

### Rapport

- **Version PDF de 10-20 pages (sans les annexes)**
- Qualité de l'analyse et de l'orthographe à prendre en compte
- exemple de contenu :
  - présentation de l'équipe,
  - présentation du projet (description du site, du contexte, motivations, objectifs, concept),
  - répartition du groupe et planning,
  - *(recherches graphiques)*,
  - architecture et gabarit du site (arborescence, schéma de la bdd),
  - tâches et fonctionnalités à implémenter *(liste des endpoints)*,

- gestion de projet (avancement, méthodes utilisées),
- rendu, satisfaction du projet,
- post-mortem
- *annexes : tout le processus de la création de la bdd, manuel utilisateur API (liste des endpoints avec: **titre, description, méthode, URL, paramètres URL/data(corps de requête)**, réponses/erreurs, exemples d'utilisation, ...)*

## Composition des groupes

### 4 à 5 IMAC1 par groupe

dont un.e chef.fe de projet

- Maîtrise de **PHP et SQL** : Fonctionnalité du code (qualité du code exigée : PSR) et de la bdd
- Côté front: requêtes HTTP fonctionnelles
- Respect de l'architecture REST
- Nombre de fonctionnalités implémentées par rapport au nombre prévu (respect des endpoints/routes)
- Participation du groupe
- Soutenance : forme, aisance, présentation du site en live
- Rôles :

	Chef.fe de projet	Développeu.r.se back
Gestion de l'équipe back	+	
Communication	++	++
Fonctionnalités implémentées	++	++
Qualité du code	+	++
<i>Design</i>	+	+

La/Le chef.fe de projet sera moins noté.e sur la qualité de son code, en contrepartie iel aura les responsabilités suivantes:

- iel sera la/le principal.e interlocut.eur.rice lors du suivi de projet
- iel s'occupera au bon déroulement du projet: bonne communication entre membres du groupe, bonne répartition des tâches, gestion du planning, gestion des possibles conflits...

Concernant le design du site, prévenir qui s'en est chargé, le graphisme peut jouer dans la note.

## Barème

Le barème est indicatif, il peut y avoir des ajustements (même si je suis obligée vous noter j'adore ce que vous faites).

### Rendu final : /10

- Rendu avant la **date limite (date à préciser)**
- Organisation du repo Git
- Qualité du code
- Qualité de la bdd
- Contenu du rapport
- Plus vous serez organisé.e.s en gestion de projet, plus vos efforts seront valorisés

### Notation individuelle : /5

- Selon le rôle et son implication dans le projet (c.f. Composition des groupes)
- Les notes hors projet (TDs)

### Soutenance : /5

- **Soutenance fin mai (date à confirmer)**
- **10 minutes de présentation + 5 minutes de questions par projet**
- Participation de tout le groupe
- Aisance de la parole
- Conformité au rapport
- Démo du site : (design,) fonctionnalités présentes, ergonomie, originalité du concept
- Donne des réponses (bonnes ou pas oser) argumentées aux questions / remarques
- Présentation positive, vendez votre produit ! N'essayez pas d'être pessimistes (lorsque le groupe bloque, il doit réussir à expliquer leur problème, comment ils ont réussi (ou pas) à le surmonter, ce qu'ils ont appris, ce qu'ils auraient dû faire, parce que ce n'est pas un drame d'échouer...)

## Planning

- **fin février** : composition définitive des groupes et publication du brief (ceci)
- **avant le 16/03 à minuit** : livraison lien du repo Git + pitch du site
- **avant le 06/04 à minuit** : livraison première ébauche du schéma de la bdd
- **avant le 13/04 à minuit** : livraison première ébauche du site en ligne
- **à partir des vacances de Pâques** : suivi des projets

- **une semaine avant la soutenance (date à confirmer) :** livraison rapport et version définitive du site
- **fin mai (date à confirmer) :** soutenance (10 min présentation + 5 min questions)

## Inspirations et Conseils

### Design :

- N'hésitez pas à faire des recherches sur Behance, Pinterest ou ailleurs sur Internet !

### Rappel de conseils Git / travail en groupe :

- Faites beaucoup de commit, et committez bien ! (N'envoyez pas de code qui ne fonctionne pas !)
- Séparez les tâches indépendantes, gardez une branche "stable" (master) à tout moment du développement. On ne travaille pas directement sur master.
- Jouez collectif : organisez-vous, concertez-vous avec les partenaires, mettez des titres compréhensifs à vos commits (pas du genre "IT WORKS" thx), suivez les avancées des autres, donnez votre avis, ...
- Parlez beaucoup entre vous ! N'hésitez pas à me poser des questions si jamais !
- Servez-vous de Trello (<https://trello.com/>) pour organiser vos tâches ("kanban")