

Grille d'évaluation Livre d'or TI2 - 2025

Compétences évaluées Partie 1 (TI du 28/04/2025)

- **Respect des consignes** : Les fonctionnalités sont conformes à la demande. Si nommés dans les consignes ou dans la grille d'évaluation, comme les noms de variables, fichiers etc... doivent être conformes à ceux demandés !
- **.git** : fork, clone, .gitignore, sauvegardes, travail remis via github
- **Wamp** : création d'un hôte virtuel vers le dossier public du test
- **PhpMyAdmin** : importation d'une DB contenant une table depuis un fichier .sql
- **MySQL (MariaDB)** : exécutions de requêtes SELECT et INSERT
- **Structure** : respect du Modèle Vue Contrôleur (MVC) en PHP
- **PHP** : Gestion du contrôleur frontal, d'un modèle et d'une vue
- **PHP** : Gestion de la connexion et des requêtes avec PDO
- **PHP** : Sécurité backend (contre les injections SQL)
- **HTML / CSS** : formulaire responsive
- **JS** : validation Front-end des champs du formulaire

Git et GitHub

Tâches	Critères	Indicateurs	Niveau de réussite	
GitHub	Création d'un fork du dossier de consignes	Pull request sur le dossier de consigne à 16h30	Les 2 points sont obligatoires	
	Mise à jour régulière du travail sur github	Utilisation de commit réguliers et/ou de branches		
Git	clone	Le dossier cloné est le fork du dossier de consigne	2 points sur 3 sont nécessaires	
	Le .gitignore empêche les fichiers non nécessaires ou dangereux d'être mis dans les sauvegardes	config.php et les fichiers non nécessaires ne sont pas intégrés dans la sauvegarde		
	upstream	Le lien vers l'upstream pointe vers le dossier de consignes. Un pull request peut être effectué		
Compétence acquise				

WAMPserver et phpMyAdmin

Tâches	Critères	Indicateurs	Niveau de réussite
WAMP	Création d'un hôte virtuel	L'origine de l'hôte virtuel pointe vers le dossier public du fork	Point obligatoire
Création de la DB (MySQL/MariaDB)	Importer une DB depuis un fichier SQL	Sélection d'un fichier .SQL Import au niveau serveur La/les table(s) sont créées et le(s) champ(s) sont corrects	Les 3 points sont obligatoires
Compétence acquise			

PHP

Tâches	Critères	Indicateurs	Niveau de réussite
Structure des fichiers	La partie fonctionnelle et structurelle du site doit suivre certains critères	Site fait en MVC procédural allégé (ici, uniquement vue(s) et modèle(s), le contrôleur étant le contrôleur frontal dans le dossier public) Utilisation d'un contrôleur frontal pour la partie publique : /public/index.php Utilisation d'un fichier de configuration hors de portée de l'utilisateur : config.php Utilisation d'une vue hors de portée de l'utilisateur: /view/guestbookView.php Utilisation de sous-dossiers frontend pour le css, les images et le javascript (dans le dossier public)	4 points sur 5 obligatoires
PHP	Formulaire	Récupération des champs via la méthode POST Appel du modèle si les champs	Tous les points sont obligatoires

		<p>sont valides, vérification et nettoyage du contenu des champs dans le modèle</p> <p>Envoi d'erreurs (false) en cas de champs non valides (non détaillé en PHP, faire en Javascript)</p>	
PHP / SQL	Lien à la DB	Récupération des constantes de connexion dans config.php	Point obligatoire
Connexion		Utilisation d'un try catch pour la connexion	6 points sur 8 obligatoires
		Connexion avec PDO	
		Mettre utf8mb4 comme jeu de caractère de la connexion	
		Arrêt du script et affichage des erreurs de connexions	
		Utilisation de la classe Exception pour la gestion des erreurs	
		La fermeture de connexion est présente à un endroit correct	
		PDO::FETCH_ASSOC est activé lors de la connexion	
		PDO::ERRMODE_EXCEPTION est activé lors de la connexion	
Lecture (SELECT)		Le(s) message(s) stockés en base de données sont récupérés	5 points sur 8 obligatoires
		Le(s) message(s) récupérés sont affichés	
		Les messages sont affichés du plus récent au plus ancien	
		Si pas de message, affichage de "pas encore de message"	
		Si on a un seul message, on affiche "Il y a 1 message"	
		Si on a plus d'un message, on affiche le nombre au format "Il y a X messages"	
		Utiliser une requête préparée	
		On vide les résultats du SELECT lorsqu'ils ne sont plus nécessaires	

		BONUS PRINCIPAL : Une pagination fonctionnelle est mise en place	
		BONUS : La date du message est formatée au format français : "Le (27/04/2025 à 10h29)"	
		BONUS : Le retour automatique à la ligne est activé dans les messages	
Ecriture (INSERT)		Tous les champs nécessaires sont présents	6 points sur 9 sont obligatoires
		Les champs sont traités pour empêcher une injection SQL et n'acceptent ni les espaces avant-arrière, ni les tags html. Les caractères spéciaux doivent être encodés, dont le ' et ". le mail est dans un format valide ainsi que les numériques souhaités ('phone' et 'postcode')	
		La requête doit être préparée	
		Seuls les champs `id` et `datemessage` dans la DB peuvent être vides dans les champs du formulaire	
		Si un champ n'est pas valide, pas d'exécution de la requête d'insertion (envoie de false)	
		Un try catch est obligatoire pour l'insertion	
		Une erreur SQL fait un die() avec le message de l'erreur	
		Les erreurs d'insertions renvoie un false dans index.php ce qui affiche dans la vue: "Problème lors de l'envoi du message" (rouge)	
		Le nouveau message est stocké en base de données et renvoie true dans index.php, ce qui affiche "Merci pour votre nouveau message" en cas de réussite (vert),	

Compétence acquise	
-------------------------------	--

HTML

Tâches	Critères	Indicateurs	Niveau de réussite
Page	Contenu	le code est placé dans la vue guestbookView.php	Les 2 points obligatoires
		Titres, image, formulaire et messages sont présents	
Formulaire	Contenu	Les champs obligatoires et optionnels sont corrects	3 points obligatoires sur 4
		Le contenu du formulaire est correct	
	Fonctionnement	Un message d'avertissement apparaît pour signaler si le message est enregistré ou non, dans les mêmes champs que Javascript : vert en cas de réussite, rouge en cas d'échec (voir Javascript et PHP)	Point obligatoire
		Les données sont envoyées au serveur (si formulaire valide)	
Posts précédents		Chaque post est bien distinct, avec prénom, date, email et contenu	Point obligatoire
Compétence acquise			

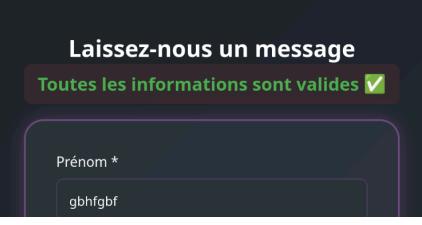
CSS

Tâches	Critères	Indicateurs	Niveau de réussite
Affichage “responsive” de la page	La mise en page est différente selon la largeur de la page	Les éléments sont positionnés correctement pour la vue “smartphone” Les éléments sont positionnés correctement pour la vue “tablette” Les éléments sont positionnés correctement pour la vue “desktop”	3 points obligatoires
Formulaire		La mise en page est soignée (alignements) Le message d'avertissement apparaît pour signaler si le message est enregistré (en vert) ou non (en rouge)	1 sur 2 obligatoires
Posts précédents		Tous les posts sont affichés avec la même mise en page (class commune)	obligatoire
Compétence acquise			

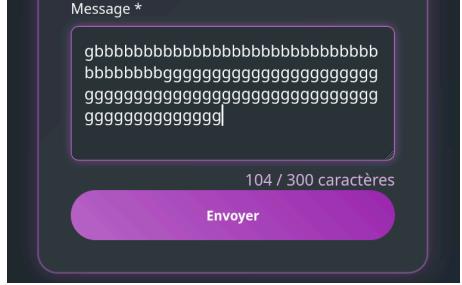
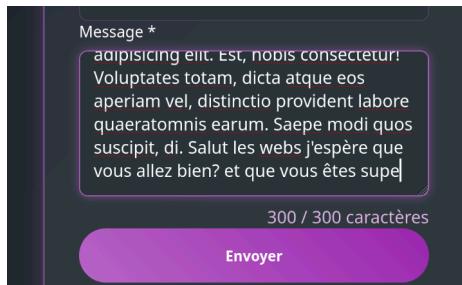
JS (validation)

Tâches	Critères	Indicateurs	Points /20
Validation Formulaire	Vérification des champs	Tous les champs sont obligatoires mais les champs: (email,code Postal,téléphone) sont à vérifier à travers leurs ID ou leurs CLASS sur le formulaire.	2 points
	Manipulation DOM	Récupérer à l'aide de leurs "ID" chaque champs à vérifier puis les mettre dans des boîtes.	3 points
	Gestion Formulaire	S'assurer, à l'aide de regex, d'avoir comme retour un booléen pour chaque champ testé.	5 points
		Pour chaque champ, une vérification doit se faire en accord avec l'expression régulière adaptée accompagnée d'une alerte, si la condition n'est pas respectée.	
	Intégration avec PHP	L'utilisateur ne peut pas envoyer le formulaire tant que les champs ne sont pas remplis conformément aux conditions mis en place par votre Javascript.	5 points
Compétence acquise			

BONUS

Tâches	Critères	Indicateurs	Points
Gestion Formulaire	<p>Un message en rouge apparaît en haut pour chaque champ qui ne respecte pas les conditions JS. </p> <p>Vert quand tous les champs remplissent les conditions et l'envoi / ou le formulaire est envoyé et c'est PHP qui affiche "merci pour votre nouveau message" en cas de réussite (vert), "Problème lors de l'envoi du message" (rouge) </p>	 	

Tâches	Critères	Indicateurs	Points
--------	----------	-------------	--------

Gestion du compteur de caractères	<p>Le champ de message doit être interactif : l'utilisateur doit voir en temps réel le nombre de caractères qu'il a saisis.</p>	 <p>Message *</p> <div style="border: 1px solid #ccc; padding: 5px; background-color: #f9f9f9;">acipisicng eit. Est, nobis consectetur! Voluptates totam, dicta atque eos aperiam vel, distinctio provident labore quaerat omnis earum. Saepe modi quos suscipit, di. Salut les webs j'espère que vous allez bien? et que vous êtes supe </div> <p>300 / 300 caractères</p> <p>Envoyer</p>	<p>2 points</p>
	<p>L'utilisateur ne peut plus rien insérer dès lors qu'il atteint la limite des 300 caractères</p>	 <p>Message *</p> <div style="border: 1px solid #ccc; padding: 5px; background-color: #f9f9f9;">acipisicng eit. Est, nobis consectetur! Voluptates totam, dicta atque eos aperiam vel, distinctio provident labore quaerat omnis earum. Saepe modi quos suscipit, di. Salut les webs j'espère que vous allez bien? et que vous êtes supe </div> <p>300 / 300 caractères</p> <p>Envoyer</p>	

Respect des consignes

Tâches	Critères	Indicateurs	Niveau de réussite
Respecter les consignes du TI	A compléter		
Compétence acquise			

Compétences évaluées Partie 2 (Présentation du 5 mai 2025)

- **FTP** : Le travail est mis en ligne
- **SQL** : Le travail utilise une base de donnée distante
- **Présentation publique technique** : Pouvoir présenter son travail de manière cohérente et pouvoir répondre aux questions plus techniques

Production :

Tâches	Critères	Indicateurs	Niveau de réussite
Le travail est mis en ligne (Production)	FTP	Les fichiers sont présents dans l'espace web fourni au stagiaire	3 points sur 4
		Les liens vers les images, css, js et pages sont corrects lors de la mise en production	
	SQL	La table 'livreor' a bien été transférée sur la base de donnée du serveur SQL distant fourni au stagiaire	
		La connexion à la base de donnée est réussie	
Compétence acquise			

Présentation publique

Tâches	Critères	Indicateurs	Niveau de réussite
Présentation publique	Présenter son travail (explication des choix, explication du code, difficultés rencontrées et solutions)	Pouvoir présenter son travail de manière cohérente	
		Le code utilisé est compris	
Compétence acquise			