Bilan du projet ENI-Encheres

# Résultat

|  |  |  |  |
| --- | --- | --- | --- |
| **Fonctionnalités** | **A faire** | **En cours** | **Fait** |
| Se connecter |  |  | X |
| S’inscrire |  |  | X |
| Modifier mon profil |  |  | X |
| Vendre un article |  |  | X |
| Lister les ventes |  |  | X |
| Faire une enchère |  |  | X |
| Remporter une vente |  |  | X |
| Version mobile |  |  | X |
| Afficher un profil |  |  | X |
| Afficher le détail d’une vente |  |  | X |
| Version desktop |  |  | X |
| Se souvenir de moi | X |  |  |
| Photo pour la vente |  |  | X |
| Annuler une vente | X |  |  |
| Supprimer mon compte |  |  | X |
| Supprimer des comptes utilisateurs | X |  |  |
| Désactiver un compte utilisateur | X |  |  |
| Pagination | X |  |  |
| Mot de passe oublié | X |  |  |
| Gestion des logs | X |  |  |
| Version anglaise | X |  |  |
| Gestion catégories |  |  | X |

Avez vous réalisé des fonctionnalités non prévues initialement ? Non

|  |  |  |  |
| --- | --- | --- | --- |
| **Critères techniques** | **A faire** | **En cours** | **Fait** |
| Respect des principes de l’architecture en couches |  |  | X |
| Gestion correcte de la navigation : retour navigateur, rafraîchissement d'une page avec formulaire, etc. |  |  | ? |
| Sécurisation de l’accès aux pages nécessitant une authentification |  |  | X |
| Gestion de la session |  |  | X |
| Factorisation du code (Servlets, JSPs, CSS...) |  |  | X |
| Utilisation efficace des contextes mémoire |  |  | X |
| Séparation HTML/CSS |  |  | X |
| Modularisation des vues (insert, fragments, templates…) |  |  | X |
| Portabilité du code (pas d’informations sur le serveur, port, ou nom d’application codées « en dur ») | X |  |  |
| Séparation effective des technologies (ex : minimum de scriptlets dans les JSP). |  |  | X |
| URI logiques : les noms réels des fichiers ne doivent pas être visibles dans le navigateur. (Ex : /seConnecter vs ConnexionServlet ) | X |  |  |
| Respecter le pattern MVC |  |  | X |
| Adaptation de la mise en page au format de l’affichage (Responsive Web Design) | X |  |  |
| Utilisation d’un pool de connexions |  |  | X |
| Pas de fuite de connexions |  |  | ? |
| Pas d’injection SQL possible |  |  | X |
| Garantir l’intégrité des données (exemple : enchérir, nouvelle vente...) |  |  | ? |

# Évaluation rétrospective :

|  |  |  |
| --- | --- | --- |
| **Critère de qualité** | **Niveau (1 à 5)** | **Comment atteindre 5 ?** |
| Maintenabilité (respect des conventions, nomenclature, utilisation de patterns, documentation…) | 3 |  |
| Robustesse (absence de défauts/bugs, sécurité, bonnes pratiques…) | 3 |  |
| Évolutivité (paramètres externalisés, utilisation de pattern…) | 3 |  |

# Réussites (ce que vous considérez comme le point fort de votre projet ou dont vous êtes particulièrement satisfaits)

Versioning et le backend.

# Difficultés rencontrées (obstacles ayant eu un impact sur l’avancement du projet)

Environnement de travail (SQL SERVER / Connexion)

# Pistes d’améliorations (comment améliorer le déroulement du projet par rapport aux pratiques mises en œuvre)

***Commencer à… (ce que nous ne faisions pas)***

***Arrêter de… (ce qu’il ne faudrait plus faire)***

***Continuer à… (ce qui s’est montré utile, efficace)***

***Plus de… (ce qui peut être amélioré)***

***Moins de … (ce qui doit être réduit)***