PROJET FINAL — Sécurité des applications web

Module : Sécurité des applications

Groupe : **4 étudiants**

Soutenance : **Vendredi prochain**

Objectif général

Vous devez **concevoir, auditer, sécuriser et présenter** une **application web complète**, en appliquant **l'ensemble des notions vues pendant le module**.

Ce projet simule une **mission réelle en entreprise** :

👉 une application existe (ou démarre),

👉 elle doit être **sécurisée sérieusement avant mise en production**.

Architecture obligatoire

Votre application doit respecter **au minimum** l'une des architectures suivantes :

Option 1 — Architecture 3-tiers

- **Front-end** (client web)
- **API back-end**
- **Base de données**

Option 2 — Microservices

- Plusieurs services back-end
- Communication via API
- Base(s) de données associée(s)

⚠️ Une simple app monolithique **n'est pas suffisante**.

Sécurisation OBLIGATOIRE (cœur du projet)

Votre application doit intégrer **concrètement** tout ce qui a été vu dans le cours.

1️⃣ Sécurité des communications

- HTTPS (TLS)
- Justification SSL / TLS
- HSTS (si pertinent)
- Headers HTTP de sécurité
- Configuration CORS maîtrisée

2️⃣ Authentification & sessions

- Système d'authentification fonctionnel
- Hashage sécurisé des mots de passe :
  - **bcrypt ou Argon2 obligatoire**
- Gestion correcte :
  - sessions ou JWT
  - expiration

    - expiration
    - cookies sécurisés (Secure, HttpOnly, SameSite)
- Protection contre :
    - brute force
    - session fixation

3️⃣ Protection contre les vulnérabilités OWASP

Vous devez traiter **au minimum** :

- Injection (SQL / NoSQL / autre)
- XSS
- Broken Access Control
- Security Misconfiguration
- Cryptographic Failures
- Vulnerable & Outdated Components

👉 Chaque protection doit être **expliquée et justifiée**.

4️⃣ Stockage sécurisé des données

- Aucune donnée sensible en clair
- Séparation claire :
    - données
    - secrets
    - configuration
- Gestion correcte :
    - variables d'environnement
    - .env

    - .env.example
- Aucun secret dans le front-end

5️⃣ Outils de sécurité (OBLIGATOIRES)

Vous devez utiliser **au minimum** :

- **SonarQube** (analyse statique)
- **OWASP ZAP** (analyse dynamique)
- **npm audit** (dépendances)

👉 Vous devez :

- analyser les résultats
- identifier les failles pertinentes
- corriger ou justifier les risques
- expliquer les faux positifs

Utilisation de l'IA (obligatoire, notée & transparente)

Vous devez inclure une **partie dédiée** expliquant :

- quels outils IA ont été utilisés
- pour quoi (code, debug, compréhension, sécurité…)
- ce que l'IA vous a fait gagner
- ce qu'elle vous a fait perdre
- les limites rencontrées

- votre regard critique

⚠️ **Transparence totale exigée**

👉 L'IA est un outil, pas un auteur.

Rapport écrit (PDF — OBLIGATOIRE)

Document structuré, professionnel, clair

Plan attendu (indicatif mais recommandé)

1. Présentation du projet
2. Architecture technique
3. Conception fonctionnelle
4. Développement de l'application
5. **Sécurisation de l'API**
    - failles identifiées
    - protections mises en place
6. **Outils de sécurité**
    - SonarQube
    - ZAP
    - npm audit
7. Gestion des secrets & configuration
8. Utilisation de l'IA (section dédiée)
9. Limites, difficultés, axes d'amélioration
10. Conclusion

👉 **Le rapport doit montrer votre raisonnement**, pas juste des screenshots.

Soutenance orale

⏱️ **10 minutes de présentation**

❓ **5 minutes de questions**

Attendus à l'oral

- Présentation claire de l'application
- Démonstration si pertinent
- Mise en avant :
    - des failles
    - des corrections
    - des choix de sécurité
- Capacité à **justifier techniquement**

Critères d'évaluation

Critère	Pondération
Architecture & cohérence	✔
Qualité de la sécurisation	✔✔✔
Maîtrise OWASP	✔✔
Usage des outils (Sonar/ZAP/npm)	✔✔
Qualité du rapport	✔✔
Présentation orale	✔
Esprit critique & IA	✔

Règles importantes

- Toute faille non traitée doit être **justifiée**
- Toute décision doit être **argumentée**
- Toute sécurité “cosmétique” sera pénalisée
- Un projet fonctionnel mais non sécurisé = **échec**