

Scientifique (transversal)

Projet Transversal (80 heures)

26_27_4IRC_07_UE4_PRJSC_040_C

ACQUIS

- Mettre en oeuvre et intégrer diverses compétences à travers un projet collaboratif
- Réaliser un projet en équipe de 4 ou 5 étudiants, en combinant les compétences en développement logiciel, réseau, IoT et sécurité pour une approche multidisciplinaire.
- Analyser et synthétiser les spécifications et le cahier des charges
- Étudier et comprendre les exigences techniques et fonctionnelles, puis synthétiser ces informations pour guider le développement du projet.
- Développer des solutions logicielles et matérielles adaptées
- Concevoir et implémenter des solutions en tenant compte des contraintes liées au déploiement en réseau, tout en respectant les bonnes pratiques de conception et de programmation orientée objet.
- Développer une application web sécurisée
- Concevoir et implémenter un protocole de transmission de données IoT sécurisé
- Concevoir, implémenter et sécuriser un protocole efficace pour la transmission des données issues des objets connectés, en garantissant la fiabilité et la sécurité des échanges.

CONTENU

- Analyse du cahier des charges
- Étudier et interpréter les spécifications fonctionnelles et techniques pour en extraire les exigences principales du projet.
- Analyse des interactions entre les différents composants (logiciel, réseau, IoT)
- Examiner les relations et dépendances entre les applications, le réseau et les dispositifs IoT pour assurer une intégration fluide.
- Définition des jalons et mise en oeuvre d'une stratégie de gestion de projet
- Planifier les étapes clés du projet, établir un calendrier de travail et adopter une méthode de gestion de projet pour suivre l'avancement et les risques.
- Conception et réalisation d'une architecture réseau
- Concevoir l'architecture réseau adaptée aux besoins du projet, en tenant compte des contraintes de performance, de sécurité et de scalabilité.
- Conception d'une chaîne IoT et définition du protocole d'échange de données
- Développer une chaîne IoT complète en sélectionnant les capteurs, les dispositifs et les protocoles de communication (comme MQTT, CoAP) nécessaires.
- Conception de l'architecture logicielle en respectant les bonnes pratiques apprises précédemment dans le cursus
- Appliquer les principes de conception orientée objet (COO/POO), les Design Patterns (DP) et les meilleures

pratiques acquises lors de la formation pour assurer la modularité et l'évolutivité de l'architecture logicielle.

- Conception et réalisation de la base de données
- Concevoir une base de données performante et adaptée aux besoins du projet (relationnelle ou NoSQL), en définissant les structures et les relations pertinentes.
- Réalisation d'une infrastructure web pour la gestion des données
- Développer une plateforme web permettant de collecter, transférer et exploiter les données générées par les objets connectés.
- Analyse, traitement et visualisation des données
- Mettre en place des outils d'analyse et de traitement des données collectées, ainsi que des visualisations claires pour faciliter leur interprétation et prise de décision.
- Sécurisation de la chaîne de transmission IOT
- Sécuriser la transmission de données IOT avec la mise en place du protocole TLS.
- Mise en place de mesures de sécurisation pour l'application développée
- Renforcer la validation des entrées afin de prévenir les attaques par injection (SQL Injection, XSS, etc.).
- Implémenter une gestion rigoureuse de l'authentification et des sessions pour garantir la sécurité des accès et éviter les compromissions.
- Intégrer des outils d'analyse de code statique (tels que SonarQube) dans le pipeline CI/CD afin de détecter les vulnérabilités potentielles dès la phase de développement.
- Réaliser des tests dynamiques de sécurité d'application (DAST) à l'aide d'outils comme OWASP ZAP ou Burp Suite pour identifier les failles en cours d'exécution.
- Sécuriser les API en mettant en place des mécanismes d'authentification robustes et des limitations de taux (rate limiting) pour prévenir les attaques par déni de service (DoS).

PRÉREQUIS

- Développement en C et Java.
- Bonne connaissance des concepts POO (classes, héritage, polymorphisme, encapsulation).
- Gestion de bases de données relationnelles (SQL).
- Réseaux (Commutation et routage).
- Développement d'applications web sécurisées.
- Cryptographie, TLS et sécurité du web et du code.

PÉDAGOGIE

ÉVALUATION

BIBLIOGRAPHIE