Attentes minimales pour les TFE en

Technologie de l’Informatique

# Commun à tous les thèmes

* **Choix du sujet** :
  + Le sujet doit être pertinent et spécifique, et doit être validé par le rapporteur. Il doit : présenter un intérêt technique, un intérêt pour un client, apporter une plus-value, une spécificité ou un positionnement par rapport à des solutions déjà existantes, et éventuellement être validé par une étude de marché ou des sondages.
  + Le travail doit représenter une charge de travail suffisante, également validée par le rapporteur (300h environ, avec analyse en conséquence).
* **Cahier des charges** : Il s’agit de la formalisation de la demande du client. Ce cahier des charges constitue la première étape indispensable à la réalisation du TFE. Il doit être montré au rapporteur dès que possible. Il contient les éléments suivants :
  + Le contexte dans lequel le TFE s’articule
  + La présentation des besoins du client (il s'agit bien de décrire des fonctionnalités et non des solutions)
    - Eventuellement sous forme de User Stories
  + Les contraintes dont l’étudiant doit tenir compte (matérielles, temporelles, …)
  + La méthodologie que l’étudiant propose de suivre pour la conception, la réalisation et la validation de son travail, ainsi que les interactions avec le client (et/ou le rapporteur)
* **Analyse de la problématique** : Une fois le cahier des charges fixé, l’étape suivante consiste à réaliser l’analyse de la problématique, à savoir :
  + l’identification des besoins techniques
  + les choix justifiés des technologies
  + la conception de la solution.

Selon la thématique, les schémas adéquats seront fournis : Use Case, UML, schémas de base de données, schémas réseaux, schémas électroniques, schémas architecturaux, diagrammes de classe, …

* **Historique du projet** : Le rapport final contiendra également le planning de la réalisation et une analyse du respect de celui-ci, le bilan des rencontres avec le client (+ éventuellement rapporteur) et l’explication de l’évolution des choix dans le temps
* **Versioning** : L’utilisation d’un outil de versionning type GitHub est attendue, et ce dès le début du projet (commit unique la veille de la remise non accepté)
* **Produit fini** : Tout code/produit doit être facilement testable par l’équipe pédagogique. L’étudiant doit se concerter avec son rapporteur pour valider cette procédure.
  + Par ex, pour TFE applicatif, l’application doit être déployable instantanément par les professeurs, via la mise à disposition d’une procédure d’installation courte (par ex. image docker).
* **Démonstration du cas pratique** : dès le début du TFE, l’étudiant doit réfléchir à la manière dont il montrera sa réalisation. Si cela ne s’avère pas possible, il doit en discuter avec son rapporteur qui pourra, le cas échéant, faire une visite sur site pour pouvoir se porter garant de la réalisation effectuée. Dans ce cas, une vidéo ou une démo simplifiée sera également préparée pour que le jury puisse se faire une idée de ce qui a été fait. Idéalement le projet doit être testé et validé par le client et déployé en production. Les étudiants sont encouragés à suivre un développement incrémental de manière à pouvoir faire des démonstrations intermédiaires du produit fonctionnel.
* **Bilan** : Le rapport final doit contenir une analyse critique du projet : points forts et points faibles, améliorations envisageables, plan pour le futur
* **Ressources utilisées** : L’étudiant doit indiquer clairement les sources de toute ressource intellectuelle ou d’exploitation (site web, code source, schémas, livres...) utilisées dans le cadre du projet, selon les conventions habituelles (bibliographie, licences logicielles, …)
* **Sécurité** : Les étudiants doivent intégrer les aspects liés à la sécurité à leur TFE, dont notamment les éléments suivants :
  + Analyse de sécurité (quelle que soit la thématique) :
    - Identification des biens à protéger (assets)
    - Quels sont les risques ?
    - Quelles sont les mesures mises en place ?
    - Quels sont les risques résiduels ?
  + Chiffrement des données sensibles (ex : site en https)
  + Prise en compte des données personnelles et du GDPR

# TFE en physique appliquée (électricité, électronique, traitement du signal, télécoms)

Le travail devrait comporter une partie matérielle (par ex. : carte hardware assemblée et/ou développée par l’étudiant avec des entrées/sorties analogiques et/ou numériques, robot, systèmes d’acquisition de signaux, ...).  
Dans la mesure du possible, une simulation préalable de la solution sera développée par l’étudiant.  
La partie matérielle doit communiquer avec une interface utilisateur (application pc, mobile, ...) via des interfaces et des protocoles de communication adaptés à l’environnement dans lequel l’application tourne.  
Il faut conditionner les signaux reçus (calibration du système pour un environnement donné, nettoyage des bruits, filtrage, amplification, …) et puis les traiter pour répondre aux objectifs de l’application finale.

En plus des attentes communes à tous les projets (voir plus haut), le TFE doit expliciter :

* le besoin recherché sous forme de cahier de charges (préciser notamment les contraintes temporelles, les contraintes énergétiques, les contraintes d’espace mémoire, les contraintes environnementales, les contraintes des temps de latence dans les transferts de données, etc.),
* l'étude succincte des possibilités existantes et la justification des choix faits (pourquoi tel ou tel composant, protocole, etc.),
* l’implémentation de la solution retenue,
* des tests pour vérifier la conformité avec le cahier des charges,
* les manquements et les améliorations possibles.

# TFE en développement

## TOUS

* Le code produit est adéquatement documenté
  + Nom d’auteur et/ou source du code utilisé : les contributions individuelles seront clairement identifiées par rapport au code réutilisé. Le cas échéant, l’utilisation de code pré-existant doit être conforme à la licence liée à celui-ci.
  + Commentaires dans le code pour expliquer la logique algorithmique quand nécessaire
  + Documentation du code (ex : Javadoc)
  + Mode d’emploi pour les utilisateurs
* Le code produit doit être “propre” et répondre aux bonnes pratiques de programmation vues dans le cursus
* Le code doit être testé à l’aide de tests unitaires

## WEB

* L’ampleur du TFE en développement Web ne se mesure pas en nombres de pages web (HTML5-CSS3), mais plutôt sur base de la taille de la DB et des services proposés sur celle-ci.
* Un TFE en développement Web doit comporter de la programmation côté client ET côté serveur.
* Les aspects « sécurité » doivent être pris en compte et implémentés du coté chien et serveur.
* Les aspects GDPR doivent être pris en compte et implémentés.
* L’architecture de l’application Web doit être documentée par un schéma identifiant les éléments interagissant (Web services, proxy, BDD, … )

## Mobile

* Justifier le choix d’une application mobile
* Justifier le choix de la plateforme et des technologies choisies
* Habituellement la synchronisation de données avec un serveur est exigée
* Application concrètement installée et installable (pas simulateur)

# TFE comportant une base de données

* Les bases de données sont documentées à l’aide de schémas entités-associations et relationnels
* Les choix du type de base de données, d’un SGBD ou d’autres techniques liées aux DB sont justifiés : SQL, noSQL, ORM, logiciels, …
* Les aspects GDPR doivent être pris en compte et implémentés

TFE comportant des aspects réseaux

* L’architecture du réseau doit être correctement documentée à l’aide d’un schéma réseau clair et légendé, et expliquée/justifiée par un texte annexe.
* Toute configuration réseau doit être claire et adéquatement commentée. En particulier, l’étudiant veillera à ce qu’il ne figure pas de directive dont il ne peut expliquer l’effet, ou de directive superflue. Il est donc fortement déconseillé de “copier/coller” des configurations existantes. Une configuration minimale mais suffisante pour réaliser les fonctionnalités requises par le besoin client et par les impératifs de sécurité doit être l’objectif à atteindre.
* Les aspects de validation, de troubleshooting et de monitoring doivent être intégrés à la réflexion. L’étudiant doit documenter :
  + La manière dont il a validé son réseau
  + La manière dont on peut identifier ou être alerté d’un dysfonctionnement
  + Les outils ou procédures à utiliser pour identifier la source d’un dysfonctionnement