



Institut Provincial de l'Enseignement technique  
Nivelles  
**I.P.E.T.**

Travail de fin d'études dans le cadre de l'option

**INFORMATIQUE 7h/S**

**ENSEIGNEMENT SECONDAIRE DE TRANSITION**

**TFE**  
**CONSIGNES et ECHEANCIER**

Professeurs responsables :  
**M. Arifi, M. Benidir , M. Lorie**

**Année académique 2023-2024**

# 1. En quoi consiste le travail de fin d'études ?

Ce projet constitue un projet commun aux cours de Laboratoire de logique, Laboratoire d'informatique – T.P.T. et Informatique.

Le projet est mené obligatoirement **par groupes (ceux-ci sont imposés)**. Le projet donnera lieu à la remise d'un **dossier final écrit** de plusieurs pages et sera noté après une **défense orale** du programme. Nous accorderons une grande importance **à la qualité de la défense orale**. Cette défense a lieu devant les professeurs de l'option assistés par un jury extérieur.

Il est imposé aux étudiants d'avoir au minimum **un référent extérieur** dans le suivi de son projet. Son rôle est primordial et important dans la conception et la réalisation du projet : il permet d'avoir un regard extérieur et émet souvent des remarques pertinentes contribuant à l'amélioration du projet.

La réalisation du projet s'étalera d'octobre à fin mai. La supervision des travaux et de leurs différentes étapes sera essentiellement réalisée par M. Benidir & M. Lorie pour la classe de 6 info 1 et M. Arifi & M. Lorie pour la classe de 6 info 2.

Vu l'ampleur de ce projet, ce dernier implique une collaboration concrète et efficace entre :

- Les différents membres du groupe ;
- Les étudiants du groupe et ses professeurs (directeurs de projets) ;
- Les étudiants du groupe et son (ou ses) référent(s) extérieur(s).

Il est fortement recommandé aux élèves de répartir équitablement leurs efforts sur tout le semestre et d'y consacrer aussi du temps à domicile. **Les seules heures de cours ne pourront suffire à boucler un tel projet !!** La gestion du temps consacré au projet et l'organisation réfléchie des différentes tâches **sont primordiales** car vous devez penser aussi à **ne pas négliger l'étude de vos autres matières !**

**Les différents groupes sont pour la classe de 6<sup>e</sup> ttr info 1:**

- **groupe 1** : Diogo\_Noa\_Felix
- **groupe 2** : Jeremy\_Ilian\_Theo
- **groupe 3** : Lillian\_Lucas\_Loukas
- **groupe 4** : Matthias\_Guyaume\_Maxime
- **groupe 5** : Nathan\_Amory

**Les différents groupes sont pour la classe de 6<sup>e</sup> ttr info 2:**

- **groupe 1** : Artem\_Lucas\_William
- **groupe 2** : Ethan\_Kaysan\_Leo
- **groupe 3** : Joachim\_Matteo\_Samuel
- **groupe 4** : Maximilien\_Kylian
- **groupe 5** : Teoman\_Jeremy\_Costa

## 2. Objectifs généraux

---

**Programmer une double application technique (application en Python et site web) d'envergure nécessitera l'emploi :**

- Des concepts de base de la programmation orientée objet.
- De bibliothèques de classes pour l'accès à des bases de données relationnelles locales ou distantes.
- De bibliothèques de classes pour l'interface graphique du programme.
- De bibliothèques de classes pour l'utilisation éventuelle de fichiers plats (.txt, .xls, .csv, ...) pour l'export des données utilisateurs/joueurs, ...
- D'un site web destiné à présenter l'application réalisée en Python (= peut être considéré comme la vitrine publicitaire de l'application en elle-même).

**Pour la détermination du degré de maîtrise, les critères suivants ont toutes leur importance :**

- Écriture d'un cahier des charges complet.
- Pertinence de l'analyse (Définitions des éléments, algorithmes (en pseudo-code), schémas de conception de la base de données (MCD + MLD), schémas U.M.L. pour la conception orientée objet, schémas textuels des classes, analyse du site web, ...)
- Fiabilité des tests (Mise en place des cas de tests).
- Optimisation des structures de données, des fichiers et des programmes (= application en Python et site web de présentation).
- Présentation et documentation de l'application :
  - La documentation interne du programme (c'est-à-dire **au sein du code, les divers commentaires**).
  - Le manuel de référence destiné au programmeur ou à tout nouveau programmeur futur sur le projet (arborescence du programme et du site web, informations sur la conception, ...)
  - Le manuel d'utilisateur (**avec exemples**) destiné à l'utilisateur de l'application en Python.
- Le degré d'autonomie atteint lors de la réalisation du projet.
- La défense orale du projet devant jury.

### 3. Méthodologie générale pour l'application

---

La méthodologie comprend **5 grandes étapes** :

➤ **SPÉCIFICATION INITIALE DU SYSTÈME :**

Moment où les analystes et les utilisateurs **élaborent les spécifications d'une application et expriment les exigences provisoires.**

Exemple :

« Dans une université, il est demandé à une société en informatique de réaliser une application permettant de gérer les notes des étudiants. Chaque étudiant de l'université souhaite avoir la possibilité d'accéder à la consultation de leurs notes à tout moment et de façon informatique. Chaque étudiant doit être identifié par son nom pour pouvoir accéder à ses notes. Les étudiants sont identifiés en début d'année et les notes sont saisies par les enseignants en cours d'année. »

Plus concrètement, **il s'agit d'une première ébauche d'un cahier des charges renfermant les besoins de l'utilisateur**, de façon plus ou moins précise, pour l'application informatique qu'il souhaite.

➤ **ANALYSE :**

Moment où les analystes étudient et reformulent **avec rigueur** les besoins issus de la conception du système en constituant des modèles. Ils doivent **collaborer avec les clients pour comprendre le problème, car son énoncé est rarement complet et correct.**

Exemple :

Dans le cahier des charges ci-dessus, concernant l'application de gestion des notes des étudiants d'une université, il manque entre autres, dans les spécifications demandées par l'utilisateur, le fait de savoir comment est envisagé le problème des homonymes entre étudiants.

Le modèle d'analyse est donc une abstraction concise et précise de l'objectif du système à développer, et non de la façon dont il sera construit. Il ne doit de ce fait, **contenir aucune décision d'implémentation, mais bien, plutôt, des choix à envisager !**

### ➤ CONCEPTION DU SYSTÈME :

Moment où l'équipe de développement met au point une stratégie de haut niveau que l'on appelle : **architecture du système**. Elle définit par cette **stratégie le découpage du système en sous-systèmes**, à partir de résultats de l'analyse et dans l'objectif de définir le choix de l'architecture. C'est aussi à ce niveau que l'on vérifie la cohérence de l'ensemble du système.

Plus concrètement, il s'agit de pouvoir répondre à toutes les questions laissées en suspens au moment de la réalisation de la seconde ébauche du cahier des charges (**concernant la base de données, l'application et le site web d'un point de vue fonctionnel et conceptuel, ...**).

### ➤ CONCEPTION DES OBJETS (DES CLASSES) :

Moment où l'équipe de développement **clarifie les structures de données et les algorithmes en y ajoutant les détails d'implantation et en tenant compte de l'environnement**.

*Exemple :*

*Dans l'exemple de l'application de gestion des notes, c'est lors de cette phase-ci que nous élaborons toutes les structures de données et les algorithmes nécessaires à chaque opération des classes Etudiant, Date, ...*

### ➤ IMPLÉMENTATION :

Moment où l'équipe de développement en charge de l'implémentation **transposent les classes et les relations définies durant la conception en concepts propres à un langage de programmation, à une base de données ou à une plate-forme**. La programmation doit être facile à réaliser car toutes les décisions difficiles sont normalement déjà prises.

Et, même si aucune architecture de conception n'est imposée pour l'application, une architecture de **type MVC sera imposée pour le site web**. De plus, l'élève veillera à **organiser le code (Application et Site) de telle manière qu'il réponde aux exigences actuelles en matière d'écriture ou « de codage », de sécurisation et de conception**.

*Exemple :*

*Dans l'exemple de l'application de gestion des notes, c'est lors de cette phase-ci que nous programmerons en Python en utilisant les concepts de l'Orienté Objet.*

## 4. Sujet

---

### Pour le choix du sujet, différentes étapes et consignes sont à respecter :

Le choix du sujet a été imposé par les professeurs d'informatique afin de répondre aux exigences liées aux différents cours d'option :

- Permettre de gérer un projet complet ;
- Permettre d'organiser un travail de groupe ;
- Permettre de mener une analyse d'un projet complexe ;
- Permettre une approche orientée objet ;
- Permettre d'élaborer des bases de données ;
- Permettre la création d'un site web statique et dynamique ;

Voici ci-dessous un récapitulatif des différents choix des différents groupes de 6<sup>e</sup> ttr info 1 :

Groupe	Liste des élèves	Sujet choisi
<b>groupe 1</b>	Diogo_Noa_Felix	<b>Monopoly Casa de Papel</b>
<b>groupe 2</b>	Jeremy_Ilian_Theo	<b>Monopoly Star Wars</b>
<b>groupe 3</b>	Lillian_Lucas_Loukas	<b>Monopoly Mauvais Perdant</b>
<b>groupe 4</b>	Matthias_Guyaume_Maxime	<b>Monopoly Star Wars Mandalorian</b>
<b>groupe 5</b>	Nathan_Amory	<b>Monopoly Builder</b>

Voici ci-dessous un récapitulatif des différents choix des différents groupes de 6<sup>e</sup> ttr info 2 :

Groupe	Liste des élèves	Sujet choisi
<b>groupe 1</b>	Diogo_Noa_Felix	<b>Monopoly Mario Kart</b>
<b>groupe 2</b>	Jeremy_Ilian_Theo	<b>Monopoly Casa de Papel</b>
<b>groupe 3</b>	Lillian_Lucas_Loukas	<b>Monopoly Mauvais Perdant</b>
<b>groupe 4</b>	Matthias_Guyaume_Maxime	<b>Monopoly Star Wars</b>
<b>groupe 5</b>	Nathan_Amory	<b>Monopoly Star Wars Mandalorian</b>

## 5. Consignes générales.

---

Les différentes étapes pour aborder un projet seront vues aux cours d'informatique au fur et à mesure de l'avancement du projet.

Ces étapes feront l'objet de consignes précises (fournies à l'aide de documents via vos équipes Teams) à devoir respecter et appliquer.

Par exemple, pour la première étape, le document se trouve :

**EquipeTeams / A\_\_Consignes / 02\_\_ElaborerCahierDesCharges**

Ces étapes seront soumises à des échéances à devoir respecter. Au terme de l'échéance, le travail réalisé sera évalué et coté par les professeurs. **Ces cotes seront notifiées au bulletin comme évaluations sommatives.** Ces évaluations doivent vous aider à améliorer votre projet au fur et à mesure en fonction des remarques faites.

Pour la réussite de votre Travail de Fin d'Etude, chaque membre de l'équipe doit **IMPERATIVEMENT** travaillé sur **CHACUNE** des parties : l'analyse, le code Python, le site Web et les Bases de Données.

Il est donc indispensable, de créer un tableau des tâches spécifiant précisément le rôle de chacun. De plus, au-dessus de chaque document (diagrammes UML, organigrammes, dictionnaires des données, ...) de spécifier la personne en charge de la réalisation et éventuellement des tests et/ou corrections à l'aide d'un tableau de métadonnées.

Lors de la défense orale, des questions seront posées sur les différents parties à tous les membres du groupe.



## 6. Consignes relatives au cours de Labo Informatique et TPT (POO Python)

---

### Analyse :

- Réfléchir à **la structure globale** de votre application et visualiser celle-ci à l'aide d'un schéma adéquat.
- Concevoir **les schémas UML** complets (liés à l'interface, au cœur du programme, ...) **en respectant la syntaxe** vue en classe et visualisant les différentes classes avec attributs et méthodes, les relations, les héritages et polymorphismes établis.
- Concevoir obligatoirement le modèle textuel de toutes les classes implémentées.
- **Concevoir les algorithmes** des parties les plus complexes : définitions des éléments et pseudo-code (en utilisant la syntaxe vue en cours).

### Implémentation :

Pour rappel, toutes les classes, attributs, méthodes, instances, modules, variables, ... définis dans vos schémas et algorithmes **doivent se retrouver dans le code de votre programme avec la même structure et les mêmes noms.**

Les règles suivantes sont à respecter lors de l'écriture du programme :

- **Programmation Orientée Objet obligatoire** avec utilisation du module tkinter / d'un module d'intégration d'une interface graphique.
- Bien structurer votre programme afin **d'optimiser votre code** (plusieurs classes obligatoirement, héritage obligatoire, modules éventuellement, un programme principal optimisé, ...).
- L'interface graphique doit contenir **de préférence** plusieurs « canevas ».
- Choisir des noms de variables, d'attributs, de méthodes, de classes, d'instances, de modules, ... explicites ne permettant aucune confusion ou conflit et respectant les conventions d'écriture utilisées au cours.
- Soigner l'interface graphique, le rendre le plus **convivial** possible avec une navigation simple et intuitive pour l'utilisateur.
- Prévoir mise en page s'adaptant si possible aux autres résolutions (responsive).

- Intégrer des commentaires à bon escient dans le programme.
- Intégrer un commentaire pour toute classe, méthode, module, etc.
- Accorder de **l'importance à la grammaire, l'orthographe et la syntaxe des phrases** apparaissant dans l'interface graphique (et donc visible pour l'utilisateur de l'application) mais aussi dans toute la documentation du code.
- Faire preuve d'autonomie et de créativité.
- **Un exécutable de votre code est souhaitable.**

Les apports personnels et la créativité contribuent de façon très significative à une bonne note du projet.

## 7. Consignes relatives au cours de Labo Logique (Base de données relationnelle)

---

Vous allez devoir réaliser **une base de données communes** pour **l'application en Python** et **le site web**.

Avant la création de vos bases de données, les schémas conceptuels (MCD) et logique (MLD) complets respectant la syntaxe vue en classe doivent être établis. Ceux-ci contiendront obligatoirement au minimum pour le MCD : les entités, l'identifiant et les attributs, les relations et leurs cardinalités et pour le MLD : les tables, les clés primaires/secondaires, les contraintes d'intégrité, etc...

Pour rappel, tous les éléments mentionnés et définis dans vos schémas conceptuel et logique **doivent se retrouver dans les tables de votre base de données avec la même structure et les mêmes noms**.

On mesurera la qualité de votre modélisation conceptuelle à l'existence de plusieurs critères :

- **L'expressivité** : traduit la richesse sémantique du schéma. Elle peut être caractérisée, par exemple, par le nombre de concepts et/ou contraintes exprimés dans le tableau.
- **L'optimisation** : tend à privilégier les schémas avec un nombre minimales de redondances.
- **La lisibilité** : consiste à évaluer la représentation graphique proprement dite.
- **La simplicité** : privilégie les schémas contenant un nombre de concepts minimum. On peut la mesurer par exemple en calculant le nombre d'entités et d'associations présentes sur un schéma.

**De plus, vous devrez prévoir que, pour chaque table, il y ait un minimum de 20 enregistrements !! Ceci afin de pouvoir faire tourner et tester l'application/le programme dès la première exécution de celui-ci avec des personnages/utilisateurs fictifs.**

Les règles suivantes sont à respecter lors de la réalisation de vos bases de données :

- **Expliciter et justifier obligatoirement chacun des choix** que vous aurez posés pour chacun des éléments présents dans votre base de données
- **Choisir des noms** d'entités/tables, d'identifiants, d'attributs, de relations **ne permettant aucune confusion ou conflit et respectant les conventions d'écriture utilisée au cours** ;
- Bien structurer votre base de données
- Soigner les schémas conceptuel et logique et les rendre le plus lisible possible ;
- **Votre base doit pouvoir être accessible facilement en dehors de l'application développée** pour le TFE.

Les apports personnels et la créativité contribuent de façon très significative à une bonne note du projet.

## 8. Consignes relatives au cours d'Informatique

---

Dans le cadre du cours d'informatique vous suivez un module de cours complet en PHP. Il vous est donc demandé de réaliser un Site Web en PHP à partir d'un modèle de conception se basant sur un schéma MVC (Modèles – Vues – Contrôleur).

De plus, il est indispensable de créer une base de données qui contiendra toutes les données reprises sur le site comme les comptes d'utilisateurs (inscription, login, password) ou des données de configuration de certains paramètres du jeu, les chemins vers les fichiers téléchargeables, ...

Le site Web se décomposera en 4 parties :

1. La partie descriptive de votre groupe (partie **statique**)
  - Dans cette section, vous devrez présenter votre groupe, fournir des informations de contact, et tout autre élément qui permettra aux visiteurs de mieux vous connaître.
2. La partie descriptive du programme (partie **statique**)
  - Dans cette section, votre tâche consiste à fournir une description complète du jeu que vous avez choisi pour votre projet. Il est essentiel de décrire en détail le jeu, y compris ses caractéristiques, son objectif, ses règles, et son contexte d'utilisation. Assurez-vous de présenter une vision complète du jeu que les visiteurs du site pourront comprendre et apprécier.
3. La partie descriptive de votre travail de fin d'étude (difficultés rencontrées, améliorations souhaitées). (partie **statique**)
  - Vous devrez expliquer les difficultés que vous avez rencontrées tout au long de votre travail de fin d'étude, ainsi que les améliorations que vous souhaiteriez apporter à votre projet. Partagez vos réflexions sur le processus d'apprentissage et de réalisation.
4. La partie interactive de votre programme (partie **dynamique**)
  - Cette section est réservée aux utilisateurs inscrits sur le site. Vous devrez créer des fonctionnalités interactives, y compris des données exemples (historique ou autre), des données paramétrables (prix du terrain, etc.), et d'autres éléments interactifs. Il est essentiel de noter que ces données devront être extraites et gérées à partir de la base de données commune au site et à l'application Python. Cette section doit être conçue pour offrir une expérience utilisateur enrichie, et seul l'accès aux utilisateurs inscrits sera autorisé.

On mesurera la qualité de votre site à l'existence de plusieurs critères :

➤ **Le point de vue organisationnel**

- ✓ Avant le développement du Site Web, il est essentiel de fournir des schémas/vues de vos différentes pages web ainsi que de la structure arborescente du site afin d'analyser au mieux les interactions entre le client, le serveur et le SGBDR.
- ✓ Respect d'une architecture semblable au MVC (= Séparer le code permettant de gérer les interactions avec la BD, le code permettant de gérer les échanges/interactions avec l'utilisateur et le code permettant l'affichage du contenu présent sur les pages.)
- ✓ Organisation pertinente des données dans la base de données MySQL, création des tables, des contraintes d'intégrités, des liaisons entre tables, des index, ...

➤ **Le point de vue visuel**

- ✓ **L'ergonomie** : traduit la richesse esthétique du site. Peut-être caractérisée par exemple par le positionnement d'éléments, de menus ou autres informations données sur le site. Il s'agira de soigner l'interface graphique du site et le rendre le plus convivial possible.
- ✓ **La lisibilité** : consiste à évaluer la grammaire, l'orthographe et la syntaxe des phrases formant le contenu du site
- ✓ **La fiabilité** : privilégie les accès et erreurs possibles **au minimum**. Le site et l'accès à sa base de données seront suffisamment sécurisés afin d'éviter tout piratage.
- ✓ Un des menus du site comprendra les rubriques suivantes : Information sur l'auteur, Cadre et but du TFE, informations sur les programmes et logiciels utilisés.

➤ **Le point de vue code**

- ✓ **La propreté** de votre code PHP, HTML, CSS, Javascript
- ✓ **Programmation Orientée Objet** en PHP **conseillée (mais pas obligatoire)**
- ✓ **Mettre des commentaires** à bon escient dans le code PHP
- ✓ Choisir des noms de variables, d'attributs, de méthodes, de classes, d'instances, de modules explicites, ne permettant aucune confusion ou conflit et respectant les conventions d'écriture utilisée au cours

➤ **Le point de vue original**

- ✓ Les apports personnels et la créativité contribuent de façon très significative à une bonne note du projet.

# 9. Consignes générales pour la présentation « écrite » du projet informatique.

---

Au terme de ce projet, un DOSSIER écrit de plusieurs pages vous sera demandé. Ce rapport devra comporter au moins les éléments suivants :

- Une page de garde complète (en-tête de l'école, titre de votre dossier, nom, prénom, année scolaire,...) → voir sur teams pour le modèle,
- Une taille de police entre 10pts et 12pts max pour les textes simples ; entre 12pts et 16pts pour les titres, sous-titres, ...
- Un interligne simple (1,5 autorisé pour les étudiants à besoins spécifiques) , un espacement entre § de 10pts , utilisation des styles,
- **Une numérotation automatique** des pages, titres, sous-titres, ... cohérentes.
- Une insertion éventuelle d'en-têtes, pieds de page, notes de bas de page pour la cohérence du document,
- Un sommaire ou une table des matières (de type « automatique » grâce à la numérotation automatisée),
- **Le cahier des charges corrigé reprenant TOUS LES POINTS (dont l'introduction)**
- Un **agenda** précis de la planification de votre travail tout au long de sa réalisation (conseil : à construire au fur et à mesure du projet),
- Une partie qui décrit **toutes les analyses concernant l'application et le site associé** : les algorithmes, le schéma conceptuel de la base de données (incluant entités, attributs, relation, cardinalité, identifiant), le schéma relationnel de la base de données, modèle textuels des classes, schémas UML, la description des tables (nom, champs, type, clé primaire soulignée, clé secondaire soulignée en pointillés), la structure globale de l'application (organisation des modules), ...
- Une liste des contraintes d'intégrité de votre base de données,
- Un mode d'emploi du programme,
- Tous les cas de tests **présentés sous la forme d'un tableau reprenant tous les cas testés** ainsi que leurs résultats (et le fait que cela a mené à des modifications ou pas – cf. point suivant),
- Les bugs rencontrés résolus ou non résolus,
- Quelques résultats de cas de tests (pas tous !) illustrant le fonctionnement du programme,
- l'avis de la (les) personne(s) de référence de votre projet,

- Mettre en annexe les codes de vos programmes (taille police 8pts autorisée pour cette partie, en interligne simple, impression recto -verso autorisée pour cette partie uniquement).
- Les codes de sécurité éventuellement implémentés dans votre programme pour son utilisation ainsi que l'adresse (URL éventuelle ou site Local) et les informations d'accès au site web (Fichiers + DB mysql),
- **Une liste de tous les fichiers constituant le programme/site web et ses satellites (images, ...) ainsi qu'un schéma des répertoires utilisés pour stocker ces fichiers,**
- Une conclusion sur ce que vous a apporté votre projet et dans laquelle vous citerez aussi toutes les améliorations que vous pourriez apporter à votre programme,
- Une rubrique « remerciements »,
- Une bibliographie complète des ressources utilisées (livres, notes, liens des sites internet, codes, ...).

**Ce rapport sera dactylographié obligatoirement.** Deux versions imprimées style « **syllabus relié** » et **une version informatisée sont exigées.** La partie code du dossier peut être imprimée en recto verso pour un gain de place (mais uniquement cette partie). Une impression couleur **n'est pas exigée.**



# 10. Suivi et échéances du projet

---

## 1. Suivi :

### Régulièrement, vos directeurs de projet :

- Feront un bilan avec vous sur l'état d'avancement de votre projet,
- Veilleront à la méthodologie utilisée et à suivre pour la concrétisation de votre projet,
- Répondront aux éventuelles questions,
- Vous aideront autant que possible dans des problèmes de conception ou de programmation pouvant survenir au cours du projet,
- Vérifieront autant que possible si vous suivez les différentes consignes imposées par le projet.

### Chaque semaine, chaque membre du groupe devra :

- Montrer qu'il a progressé de façon significative dans l'avancement de son projet informatique (dès la correction définitive du cahier des charges),
- Faire preuve de réflexion et d'analyse, d'un esprit constructif et créatif,
- Montrer qu'il est acteur de son projet (importance de la motivation !),
- Faire preuve d'initiative dans la conception et la réalisation de son projet,
- Faire preuve de persévérance face aux éventuelles difficultés ou « bugs » jalonnant son projet,
- Adopter une bonne méthode de travail afin d'être efficace dans la réalisation de son projet

### A intervalle régulier,

- Chaque étudiant du groupe sera évalué sur sa participation active à l'élaboration et l'état d'avancement du TFE.

### L'étudiant peut aussi poser des questions à ses promoteurs via leur adresse email :

[younes.arifi@enseignementbw.be](mailto:younes.arifi@enseignementbw.be)

[mourad.benidir@enseignementbw.be](mailto:mourad.benidir@enseignementbw.be)

[frederic.lorie@enseignementbw.be](mailto:frederic.lorie@enseignementbw.be)

- **Du 06 au 08 novembre** : stage interne pour permettre de débiter le projet et d'entamer le cahier des charges
- **Fin février (2 jours à spécifier)** : stage interne pour permettre de débiter la programmation du projet.
- **Le dimanche 23 mai 23h59** : remise définitive du rapport écrit et informatisé du projet (dossier écrit, code du programme, code du site web et tous les supports demandés).
- **La semaine de révision** : préparation de votre défense orale devant jury.
- **Pendant la session d'examens (date fixée ultérieurement)** : défense orale du projet devant jury.

**Tout retard lors d'une échéance (remises intermédiaires des travaux demandés, schémas, ...) sera sanctionné d'un zéro !!!**

**Tout étudiant n'ayant pas remis le rapport écrit et informatisé avec code du projet à la date précisée ne pourra présenter la défense orale et obtiendra zéro à l'entière de la cotation du TFE (cotation qui constitue la cote d'examen dans chaque cours d'option informatique).**

# 11. Sauvegarde du Projet informatique : Programme et Site Web.

---

**Pour éviter les pertes de fichiers, il convient d'effectuer régulièrement diverses sauvegardes.**

Il faut donc effectuer plusieurs copies de son travail :

- **Après chaque cours et chez vous** sur clé USB ;
- Sur votre PC personnel ;
- Sur votre disque externe chez vous si vous en possédez un.
- Via un Cloud sécurisé personnel (**disponible sur le compte de l'école**) éventuellement.
- Sur un espace d'hébergement gratuit

**Les élèves sont entièrement responsables de la sauvegarde et de l'impression de leur projet.** En particulier, la défaillance ou perte d'une clé USB, d'un problème avec votre PC personnel ou imprimante, disque dur externe, Cloud, ... ne peut être considéré comme un événement imprévisible.

## 12. Document et support à rendre à l'échéance du projet.

---

- **le rapport final écrit relié, dactylographié complet** et respectant les consignes exigées **en 2 exemplaires** ;
- **un support informatisé sous forme de clé USB de petite capacité (en 2 exemplaires) reprenant :**

### **Partie écrite :**

- Le rapport écrit complet du TFE ;
- Les différents schémas et algorithmes;
- La documentation et logiciels ou bibliothèques annexes éventuelles

### **Partie Programme :**

- Le code du programme ;
- Les codes de sécurité (mot de passe) éventuellement implémentés dans votre programme ;
- Tous modules complémentaires et fichiers utilisés nécessaires à l'exécution du programme ;
- Les fichiers contenant les requêtes SQL de création de la base de données et des tables ;
- Votre base de données de test (comprenant au strict minimum 20 enregistrements par table)!
- Un exécutable si possible ;

### **Partie Site Web :**

- Votre site Web complet : Tous les fichiers sources (.html, .php, .css, .js, .jpg ou autres, ...), la base de données (sous forme d'un fichier .sql contenant les requêtes de structures et données)
- Un accès à votre site web local ou en ligne si possible (adresse et informations nécessaires : login, mdp...) ;

# 13. Préparation à la défense orale- Grille de cotation du projet informatique (écrit et oral)

---

Cette partie vous sera transmise en fin de deuxième semestre 2022-2023 mais cependant il peut déjà être annoncé que les compétences suivantes seront examinées :

- ✓ La mise en œuvre d'une recherche cohérente sur un sujet validé par les directeurs de projet :
  - Cahier des charges
  - Analyse
  - Réalisation

Et ce tant pour l'application que le site associé.

- ✓ La maîtrise de l'analyse, de l'implémentation de l'application dans une conception orientée objet, de la base de données et du site web associé ;
- ✓ La capacité à planifier, à organiser les différentes tâches au sein du groupe et tout au long du temps imparti à la réalisation du TFE ;
- ✓ L'autonomie et la capacité à réfléchir sur les difficultés rencontrées lors des différentes étapes du travail ;
- ✓ La manière de défendre oralement le projet en utilisant les techniques de communication appropriées ;
- ✓ La façon de présenter un dossier écrit en respectant les consignes imposées ;
- ✓ La capacité à respecter les consignes et l'échéancier.

De façon plus pointue, le degré de maîtrise sera évalué suivant les compétences suivantes :

- Le degré de cohérence de l'analyse et du développement réalisé tant au niveau de l'application, de la base de données que du site associé.
- La précision et la clarté dans l'expression orale et écrite.
- Le niveau de sens critique.
- La créativité et l'autonomie.

**La réception de ce document est à indiquer dans le journal de classe à la page « communications aux parents ». Les parents ou l'élève majeur signeront cet accusé de réception dans le journal de classe.**