

Objectifs Pédagogiques

Ludovic Saint-Bauzel

January 25, 2016

Contents

1	Projet	2
1.1	Document Cadre (contractuel)	2
1.1.1	Rédiger un CdC	2
1.1.2	Rédiger un avant projet	3
1.1.3	Rédiger une conception détaillée	4
1.2	Fabrication : Capacité à réaliser un prototype	6
1.2.1	Réalisation Technique	6
1.2.2	Documentation technique	9
2	Travail	10
2.1	Capacité à travailler en autonomie dans un cadre semi-professionnel	10
2.1.1	Savoir être	10
2.1.2	Communiquer sur sa production intellectuelle	11
2.2	Capacité de travail en groupe	11
2.2.1	Organisation des autres	11
2.2.2	Utilisation d'outils de partage, versioning, backup du travail	12
2.2.3	Capacité d'empathie du groupe	12
2.3	Transfert de compétences	12
2.3.1	Transférer ses connaissances à un problème nouveau	12
2.3.2	Autoformation dans les projets	13
2.4	Démarche de dépannage	13
2.4.1	Définition	13
2.4.2	Défaillance électronique	13

2.4.3 Défaillance logicielle	14
--	----

1 Projet

1.1 Document Cadre (contractuel)

1.1.1 Rédiger un CdC

- Analyse Fonctionnelle

- Activité

L'étudiant doit être capable de lister toutes les fonctionnalités/contraintes du projet. Il doit être capable de proposer une estimation des priorités de manière juste.

Juste Présente les priorités qui permettent de réaliser le projet en ayant des étapes intermédiaires fonctionnelles. Si on ne fait que les priorités les plus haute d'abord est ce que le projet tiens la route.

- Critère

- * Très Bien

L'étudiant établit une liste toutes les fonctions et les contraintes d'un objet technologique système embarqué L'étudiant n'a pas proposé de fonctions ou contraintes inutiles ou inadaptées

- * Acceptable

- L'étudiant à lister les fonctions principales et les contraintes principales.
- L'étudiant n'a pas mis de fonctions ou contrainte inadapté pour décrire l'objet

- * Non Acceptable

- L'étudiant oublie des fonctions et/ou contraintes principales, nécessaire à une description minimale du projet. (Oublie de la contrainte étanche pour un produit prévu pour aller dans l'eau)
- L'étudiant propose des fonctions ou contraintes qui ne sont pas adapté à l'analyse de l'objet.

1.1.2 Rédiger un avant projet

- Solution conceptuelle :
 - Activité
Chaque étudiant doit être capable de dessiner un synoptique commenté répondant à un cahier des charges. Il doit être capable de positionner cette solution parmi un ensemble de solutions par une argumentation Scientifique/Technique.
 - Critère
 - * Très bien
 - Les synoptiques permettent rapidement de comprendre les solutions envisagées (Les légendes sont claires, les couleurs bien choisies, les noms choisis clairs)
 - Toutes les informations techniques sont présentes et précises (transmission bluetooth)
 - * Acceptable
 - Les synoptiques permettent de comprendre les solutions envisagées mais les schémas ne sont pas clairs car les légendes et/ou les couleurs et/ou les noms et/ou les couleurs sont peu clairs ou mal choisies
 - Toutes les informations importantes sont présentes.
 - Des informations techniques manquent ou peu précises (ex transmission sans fil)
 - * Non-Acceptable
 - Les synoptiques ne permettent pas de comprendre les solutions envisagées.
 - Une information importante manque
- État du marché : Argumentation scientifique/Technique
 - Activité
Face à une question prospective technologique (quel est le meilleur capteur pour la température), l'étudiant doit être capable de proposer une méthodologie systématique pour comparer les solutions possible.
 - Critères
 - * Très bien

- Aucun élément objectif de comparaison n'est inadéquat (en trop)
 - * Acceptable
 - Il y a des éléments objectif de comparaison des solutions
 - Quelques élément objectif de comparaison sont inadéquat (en trop, inutiles, évidents. . .)
 - * Non-Acceptable
 - Il n'y a pas d'éléments objectif de comparaison des solutions ou les éléments proposés sont inadéquat
- Estimation des coûts
 - Description

Face à une solution technologique combinant des technologies, l'étudiant doit être capable de proposer une estimation de coût réaliste. Prenant en compte des aléas de prix, incluant des petites dépenses en appliquant un coefficient de sécurité ou un budget de fonctionnement.
 - Critères
 - * Très bien

Le coût est basé d'une liste exhaustive des éléments coûteux et d'une estimation des éléments complémentaires qui permet une estimation avec une erreur inférieure à 10%
 - * Acceptable

Le coût est basé d'une liste exhaustive des éléments coûteux et d'une estimation des éléments complémentaires qui permet une estimation avec une erreur inférieur à 30%
 - * Non-Acceptable

Le coût se trompe du coût potentiel de plus de 30%. Le calcul oublie des éléments coûteux.

1.1.3 Rédiger une conception détaillée

- –

Le groupe doit pouvoir l'ensemble des documents permettant la fabrication, commandes pour la réalisation finale.
- Définir une architecture logicielle
 - Critères

- * Très Bien
 - L'architecture logicielle présentent tous les éléments majeurs et permet de comprendre les interactions entre les éléments dans une utilisation standard et dans les cas particuliers.
 - une étude des librairies permet d'évaluer de manière réaliste l'utilisation des librairies extérieures.
- * Acceptable
 - L'architecture logicielle présentent l'essentiel des éléments majeurs et permet de comprendre les interactions entre les éléments dans une utilisation standard.
 - une étude des librairies extérieures existe.
- * Non-Acceptable
 - L'architecture logicielle oublie des éléments majeurs
 - Il n'existe pas d'étude des librairies extérieures.
- Définition

éléments majeurs ils sont constitués de :

 - * Algorithmes utilisés
 - * Les protocoles de communication utilisés
 - * Interactions entre les bloque (Données, Types)
 - * Prototypes des fonctions, classes
 - * Scénarios d'utilisation
 - * Schéma des Interfaces Graphiques
 - * Librairies utilisées
- Définir une architecture matérielle
 - Critères

Éléments majeurs:

 - * Les protocoles de communication utilisés
 - * Interactions entre les bloque (Données, Types, tensions)
 - * Algorithmes utilisés et leur taille en portes
 - * Les schémas électriques mis en oeuvre
 - * Scénarios d'utilisation
 - * Liste des composants pour la réalisation
 - * Très Bien

- L'architecture matérielle présentent tous les éléments majeurs et permet de comprendre les interactions entre les éléments dans une utilisation standard et dans les cas particuliers.
- Une preuve de faisabilité physique est fournie par la simulation, l'étude théorique permet de valider tous les cas possibles.
- * Acceptable
 - L'architecture matérielle présente l'essentiel des éléments majeurs et permet de comprendre les interactions entre les éléments dans une utilisation standard.
 - Une preuve de faisabilité physique est fournie par la simulation, l'étude théorique permet de valider des cas extrêmes.
- * Non-Acceptable
 - L'architecture matérielle oublie des éléments majeurs
 - Il n'existe pas preuve de faisabilité
- Écrire les procédures de recettes permettant de vérifier les différentes réalisations.
 - Critères
 - * Très bien
 - Elle permettent une exploration complète de tous les cas d'utilisation de la solution
 - * Acceptable
 - Elle permettent une exploration de tous les cas d'utilisation normaux de la solution
 - * Non-Acceptable
 - Elle oublie au moins un cas d'utilisation normal de la solution

1.2 Fabrication : Capacité à réaliser un prototype

1.2.1 Réalisation Technique

- – L'étudiant doit avoir atteint un niveau d'expertise dans l'un des objectifs ci-dessous. Un niveau d'expertise veut dire être capable en total autonomie dans l'objectif. Il est capable de faire un exemple simple de la compétence à la vitesse d'un expert (réaliser une carte composée de

5 composants sur une carte en 1/2 journée, programmer un firmware pour un micro-contrôleur combinant trois modules et des composants extérieurs sur labdeck en 1/2 journée, réaliser une interface graphique gérant un flux entrée et un flux de sortie au moins et ayant une interface utilisateur en 1/2 journée). Dans les autres savoir faire, il doit être capable de proposer une procédure de réalisation indiquant les documents, outils à utiliser.

- Réaliser carte électronique

- Utiliser un logiciel de conception de carte
- Être capable d’appliquer les règles de conception
 - * Plan de masses
 - * Boucles de masse
 - * Antennes
 - * Alimentation des composants
- Critères
 - * Très bien
 - L’étudiant est capable de réaliser la tâche dans un temps comparable au temps nécessaire pour un expert (15 min pour 5 composants).
 - * Acceptable
 - L’étudiant est capable dessiner dans le logiciel un schéma PCB d’un circuit qui connecte au moins 5 composants et qui respecte les règles de conception de carte électronique.
 - * Non-Acceptable
 - L’étudiant est capable dessiner dans le logiciel un schéma PCB d’un circuit qui connecte au moins 5 composants et qui respecte les règles de conception de carte électronique.

- Programmer un firmware

- Écrire un code fonctionnel dans une carte
- Charger le code dans la carte
- Critères
 - Liste des modules standard d’un microcontrôleur :
 - * PWM,

- * IRQ,
- * GPIO,
- * Timer,
- * ADC,
- * DCA
- * Très Bien
 - Écrire un code fonctionnel dans une carte qui utilise tous les modules de la carte.
 - Compiler le code
 - Charger le code dans la carte
- Acceptable
 - * Écrire un code fonctionnel dans une carte qui utilise au moins 3 modules de la carte parmi la liste des modules standards.
 - * Compiler le code
 - * Charger le code dans la carte
- Non-Acceptable
 - * Écrire un code non fonctionnel ou qui utilise moins de 3 modules standard
 - * N’arrive pas à compiler un code
 - * N’arrive pas à charger le code dans la carte
- Programmer une interface de démonstration sur ordinateur
 L’étudiant doit être capable de produire un programme structuré pour permettre la modularité requise à la reprise du code produit. En fonction du langage l’implémentation peut avoir un sens différent, par exemple :
 - en C** Librairie et main appelant des fonctions.
 - Fournir une API avec une explication de la librairie métier
 - Fournir une procédure de test du code
 - en C++** Classes métiers séparés des codes/classe d’application
 - Fournir une description du code avec les classes; hiérarchies,
 - Fournir une procédure de test du code
 - ...

- Critères
 - On considérera comme test unitaire un code qui fait au moins un test par fonction.
 - * Très Bien
 - L'étudiant est capable d'écrire un code de test unitaire de qui accède à toutes les fonctions et vérifie les assertions de fonctionnement du code.
 - * Acceptable
 - L'étudiant est capable d'écrire un code test unitaire de qui accède à toutes les fonctions externe (de l'api) et verifie qu'elle passent (s'exécute sans faire d'erreur segfault ou autre).
 - * Non-Acceptable
 - Le code d'évaluation par test unitaire n'est pas exhaustif par rapport à l'api.

1.2.2 Documentation technique

- –
 - Le groupe d'étudiant doit être capable de produire à la fin du module des documentation techniques qui sont comparable aux documentations techniques comparable en forme et contenu à un document produit par les entreprise du domaine (Datasheet Arm, API Boost...).
- Datasheet
 - Critères
 - * Très bien
 - Une datasheet bien documenter doit fournir les informations d'utilisation standard (Tableaux de vérité, algorithmes, chronogrammes), Les information électrique (Tension d'utilisation)
- Rapports d'analyse
- Commentaires de code
- Fichier README ou autre document de prise en main du projet

2 Travail

2.1 Capacité à travailler en autonomie dans un cadre semi-professionnel

2.1.1 Savoir être

- Autonomie

L'étudiant doit être capable de décrire les points explorés autour d'un sujet qui fait question. Un étudiant autonome est un étudiant qui lorsqu'il pose une question, elle a été circonscrite par différentes explorations dans différentes directions. Cette question est ainsi contextualisée (les hypothèses et les conséquences sont identifiées) et la réponse à cette dernière doit permettre d'avancer significativement dans le travail.

- Respect des contraintes horaires

L'étudiant doit être capable de fournir un document dans des délais courts.

- Capacité d'adaptation

L'étudiant doit être en capacité à faire évoluer son ton (vouvoiement), son niveau de discours (simple pour le client, technique pour les tuteurs techniques...) en fonction de la situation et de l'interlocuteur.

- Enthousiasme

L'étudiant doit savoir se montrer enthousiaste motivé dans et par son travail. Il doit chercher à motiver ses collègues plutôt que de participer à un sentiment de morosité.

- Vestimentaire

Un étudiant ingénieur doit être capable d'adapter sa tenue vestimentaire à la situation. Il montrera sa capacité d'adaptation vestimentaire s'habillant en tenue professionnelle lors des soutenances et rendez-vous devant des personnes externes (client, visite...).

- Écologie

Un étudiant qui est capable d'utiliser des outils ou utilitaires et d'expliquer pourquoi cela permet de réduire l'impact écologique de ses actions quotidiennes.

2.1.2 Communiquer sur sa production intellectuelle

- —
Chaque étudiant doit être capable de communiquer sur son travail individuel.
- Oralement
Au cours du module, chaque étudiant doit être en capacité à présenter synthétiquement (environ 2 min), le projet dans lequel il travaille, le rôle de la tâche qu'il a actuellement et l'état d'avancement de cette dernière ainsi que la date théorique de fin.
- Écrit
Chaque étudiant doit être capable d'écrire synthétiquement (4-20 lignes) les actions qu'il a entreprises dans une journée de travail.
- Prouver
Par l'intermédiaire d'un journal individuel, chaque étudiant doit être en capacité de démontrer les apports qu'il a eu dans le projet à n'importe quel moment du projet. Cet outil de démonstration peut être matériel ou logiciel. Il doit permettre de garder une trace aussi des travaux en groupe en informant qui a travaillé sur quoi.

2.2 Capacité de travail en groupe

2.2.1 Organisation des autres

- Rendre compte pour le groupe
Chaque étudiant du module lorsqu'il est en charge du groupe doit être en mesure de rendre compte de l'état d'avancement du projet, des risques du projet et capable de proposer des actions si le risque devient présent.
- Organisation du travail
L'étudiant doit être capable de définir les actions de chacun pour une période donnée.
- Capacité de projection du projet
Un étudiant en charge d'un groupe doit être capable d'énumérer les risques qui pourraient empêcher la finalisation de ce dernier. Face à cette énumération il doit être capable de proposer des scénarios alternatifs de déroulement du projet pour mener à bien le projet. Mener à bien le projet: Avoir un client satisfait du résultat du projet. La

satisfaction est définie principalement par la réalisation des objectifs fixés en concertation avec le client

- Maintenir à jour les tableaux de bords
L'étudiant doit être capable de demander des comptes et de maintenir à jour les tableaux de bords du projet afin de garder une vision toujours à jour de l'état d'avancement du projet.

2.2.2 Utilisation d'outils de partage, versioning, backup du travail

Le groupe doit se munir d'outils de travail collaboratif, chaque étudiant doit être capable de montrer le travail de tous les membres du groupe en terme de production intellectuelle. Les outils conseillés sont git, svn. Possible mais moins performants: dropbox, drive, nomenclature de nommage de fichiers/dossiers. . .

2.2.3 Capacité d'empathie du groupe

Chaque membre du groupe doit être capable d'expliquer les actions des autres personnes et de donner une vision simple des difficultés du moment.

2.3 Transfert de compétences

2.3.1 Transférer ses connaissances à un problème nouveau

La compétence est considérée comme acquise si chaque étudiant est capable de nommer le cours et ses concepts principaux qui lui servent pour effectuer la tâche qu'il a à réaliser, que ce soit un travail de :

- Électronique Numérique
- Électronique Analogique
- Programmation
- Simulation
- application de calculs
- Rédaction de document

2.3.2 Autoformation dans les projets

Chaque étudiant doit être capable d'expliquer le fonctionnement de tous les outils utilisés par le groupe. Il doit être capable d'utiliser l'outil pour explorer la réalisation faite dans le projet.

2.4 Démarche de dépannage

2.4.1 Définition

Face à une défaillance technique rencontrée dans le projet ou énoncée par un encadrant, l'étudiant doit être capable de proposer une hypothèse d'origine de la défaillance et une procédure de test utilisant les outils maîtrisés permettant de vérifier cette hypothèse.

2.4.2 Défaillance électronique

- Définition

Face à une défaillance électronique rencontrée dans le projet ou énoncée par un encadrant, l'étudiant doit être capable de proposer une hypothèse d'origine de la défaillance et une procédure de test utilisant les outils maîtrisés permettant de vérifier cette hypothèse.

- Sondage hardware

L'étudiant doit être en capacité à utiliser les ports de debuggage mis à disposition par la carte manipulée dans son projet: sondes JTAG, onelink, i2c, break-point d'exécution... pour sonder l'état de son système électronique.

- Oscilloscope

L'étudiant doit être en capacité à utiliser un oscilloscope de manière avancée pour observer le comportement d'un système électronique numérique ou analogique. Utilisation avancée sous entend être capable de :

- synchroniser des signaux,
- observer des trames de protocoles série
- observer des signaux sur des fronts (trigger)
- être capable d'enregistrer les données obtenues

2.4.3 Défaillance logicielle

Face à une défaillance logicielle rencontrée dans le projet ou énoncée par un encadrant, l'étudiant doit être capable de proposer une hypothèse et de choisir de manière pertinente de vérifier l'hypothèse par le choix d'une méthodologie (ex. test unitaire...) ou d'un logiciel d'analyse. Il peut choisir dans les logiciels au moins entre gdb (ddd), valgrind. On entend pertinent par l'efficacité de la méthode proposée.