

**Suivi de projet,
Micro-management
et
Utilisation de la forge Redmine.**

Table des matières

1. Réunions techniques et micro management.....	3
1. Objectifs :.....	3
2. Support.....	3
Cette liste sera complétée lors de la réunion. Aucun point ne peut être retiré, les points n'étant plus pertinents devront être barrés après justification lors de la réunion. Une remarque justifiant les choix sera spécifiée dans la colonne adjacente.....	
3. Planification.....	4
4. Encadrement :.....	5
5. Organisation et gestion.....	5
6. Temps imparti.....	5
7. Déroulement :.....	6
8. Cohérence avec la forge :.....	6
9. Influence sur la note de projet :.....	6
2. Utilisation de la forge.....	7
1. Confidentialité.....	7
2. Définition de jalons communs à tous les projets :.....	7
3. Constitution d'un projet.....	7
4. Décomposition fonctionnelle du projet :.....	8
5. Intégration à la forge.....	8
6. Chronologie des travaux et Gantt.....	9
7. Suivi du développement.....	10
8. Utilisation de la forge lors des revues de projet.....	10
9. Différents types de demandes.....	10
a. Fonctionnalité.....	10
b. Tâches.....	10
c. Demandes de remontée d'erreur (bug).....	10
d. Demandes de support :.....	11
10. Utilisation du dépôt.....	11
a. Création du dépôt initial.....	12
b. Mise en place d'un nouvel élément (programme, carte) :.....	12
11. Séances de travail :.....	13
12. Documentations techniques.....	13
13. Fichiers divers.....	13
14. Maintenance.....	13
15. Première séance.....	13
16. Validation de la cohérence du dépôt.....	14

1. Réunions techniques et micro management

1. Objectifs :

Le début du projet est souvent une période très complexe à gérer pour un étudiant ingénieur. Pourtant, les résultats finaux dépendent fortement de la pertinence des choix faits lors de ces premières séances. Il faut en particulier assurer une montée en compétences techniques très rapide afin de s'assurer que les choix futurs seront cohérents et atteignables.

Afin de vous aider à anticiper au mieux les challenges technologiques et humains que vous aurez à gérer, nous vous proposons d'organiser des réunions techniques de micro-management.

L'objectif de ces séances est de vous guider lors de la phase de démarrage du projet, par la suite et en fonction de l'autonomie que vous aurez acquise ces réunions devraient s'espacer jusqu'à coïncider avec les revues de projet.

2. Support

Lors de cette réunion, le groupe d'étudiants présente sur le vidéoprojecteur un fichier tableur qui doit être versionné.

Vous avez en charge la forme du fichier, il devra cependant comporter les éléments suivants

Pour chaque étudiant : Le nom de l'étudiant, une date de début de l'action en cours, la date de fin et une colonne récapitulant la durée actuelle. Une description de l'action, et si possible l'action suivante à effectuer.

Une To Do list (liste des choses à faire) contenant l'ensemble des points qu'il faudra traiter.

Cette liste sera complétée lors de la réunion. Aucun point ne peut être retiré, les points n'étant plus pertinents devront être barrés après justification lors de la réunion. Une remarque justifiant les choix sera spécifiée dans la colonne adjacente.

3. Planification

Ces réunions auront lieu environs une semaine sur 2 et seront obligatoires pour tous les projets.

Ces séances dureront exactement 2H. Il vous appartient de définir l'ordre de passage, si aucun groupe ne se présente, après 5 minutes nous considérerons la séance comme terminée.

Année	Séance	Durée	Contenu	Jalon
Ge4a	Séance 1	2h	Etat des lieux, actions prévues	
	Séance 2	2h	Etat des lieux, définition du jalon à venir	
	Séance 3	4h	Etat des lieux, validation des résultats obtenus, actions	Validation
	Séance 4	2h	Etat des lieux, définition du jalon à venir	
	Séance 5	4h	Etat des lieux, validation des résultats obtenus, actions	Validation
Ge5a				
	Séance 1	2h	Etat des lieux, actions prévues	
	Séance 2	2h	Etat des lieux, définition du jalon à venir	
	Séance 3	4h	Etat des lieux, validation des résultats obtenus, actions	Validation
	Séance 4	2h	Etat des lieux, définition du jalon à venir	
	Séance 5	4h	Etat des lieux, validation des résultats obtenus, actions	Validation
	Séance 6	2h	Etat des lieux, définition du jalon à venir	
	Séance 7	4h	Etat des lieux, validation des résultats obtenus	Validation

4. Encadrement :

Lors de chaque séance de projet, les enseignants seront disponibles en salle D007 et D108. Seront présent les enseignants impliqués en projet en fonction de leur disponibilité (M. Lengagne, M. Pasquier, M. Landrault, M. James, M. Smaali, M. Laffont ...),

5. Organisation et gestion

L'organisation, la gestion du temps et des informations fournies sont intégralement sous votre contrôle, vous êtes les acteurs principaux de ces réunions. Les enseignants sont des supports techniques. L'objectif secret de ces réunions étant de rendre ces derniers disponibles et d'assurer une cohérence dans leurs conseils.

Les autres enseignants participeront en fonction de leur disponibilité et de leur emploi du temps, vous pouvez cependant planifier une réunion à n'importe quels créneaux horaires en fonction des besoins et des disponibilités de chacun.

6. Temps imparti

Le temps imparti est fixe et devra être respecté. Toute dérive étant pénalisante pour les groupes suivants, nous vous demandons d'être extrêmement rigoureux sur la gestion du temps.

Le temps sera décompté à partir de votre entrée dans la salle. Dès la fin du temps imparti le groupe suivant sera invité à s'installer. Certains enseignants pourront cependant quitter la réunion pour travailler avec vous sur certains points plus en détail.

La durée est de 10 minutes par projet dont 5 minutes maximum de présentation.

7. Déroulement :

Chaque étudiant fait un bilan sur son action en cours. Détaille les résultats obtenus. Indique les difficultés rencontrées. En aucun cas, cet exposé ne peut durer plus de 5 minutes.

Lorsqu'une action est terminée, l'action suivante est choisie dans la liste d'actions à mener.

Une action doit être calibrée pour une période de 15 jours (travail personnel inclus).

Il appartient aux enseignants d'essayer de respecter cette durée. En concertation avec l'étudiant, le sujet précis de l'action est défini pour pouvoir être traité en 15 jours.

Dans le cas contraire, elle est prolongée pour 15 jours complémentaires. Dans ce cas, son sujet peut être étendu.

La période peut être décrétée « de consolidation » dans ce cas aucune nouvelle action ne sera prise, mais le travail déjà réalisé est rationalisé (validé en profondeur, débuggé, versionné correctement, testé dans des cas réels...)

Ces périodes de moindre implication permettent d'anticiper des périodes où l'étudiant aura moins de temps pour travailler sur le projet.

Si lors de ces discussions des actions complémentaires doivent être menées elles seront ajoutées immédiatement à la liste des actions à effectuer. Un étudiant du groupe est en charge de reporter rapidement sur le document projeté les modifications signalées par les enseignants.

8. Cohérence avec la forge :

L'ensemble des actions traitées peuvent être considérées comme des micro-tâches. Les intégrer à la forge est possible, mais risque d'être fastidieux.

Cependant, toutes les actions doivent être liées à une des tâches de la forge. La finalisation d'une action a donc une influence directe sur le taux de finalisation d'une tâche. Ceci vous permettra donc de mieux définir l'évolution de votre projet, sur la forge.

9. Influence sur la note de projet :

Tout jalon non validé aura une influence directe sur la note finale de projet.

Les jalons ont les valeurs indicatives suivantes en points :

Ge4a : 1^{er} : 2points, 2ième 3 points,

Ge5a : 1^{er} : 2 points, 2ième : 3 points et 3ième 5 points.

2. Utilisation de la forge

Ce document décrit l'ensemble des opérations à mener au cours de séance afin d'assurer une bonne communication, et un bon suivi de votre projet. Ces méthodes sont aujourd'hui omniprésentes dans l'industrie, leur maîtrise est de plus en plus critique au sein des grands groupes. Pour les PME, la maîtrise de ces procédures est un réel avantage.

1. Confidentialité

Avant tout ajout d'informations sur la forge, veuillez vous assurer du niveau de confidentialité demandé par votre client.

2. Définition de jalons communs à tous les projets :

Date	Jalon (Version)
Mi-mars	Revue préparatoire
Fin avril	Revue d'appel d'offres
Mi-Septembre	Date butoir de proposition des sujets de sous-traitance
Mi-septembre	Revue de lancement
Mi-novembre	Revue d'avancement
Fin décembre	Revue de fin de projet
Début janvier	Soutenance
Mi-janvier	Fin des projets

Les jalons concernant la sous-traitance sont des jalons « secondaires », il convient cependant d'anticiper aux mieux les actions à mener pour ces échéances.

Sous le gestionnaire de projet utilisé (Redmine) un jalon s'appelle une version.

3. Constitution d'un projet

Si nécessaire, pour chaque projet, créer un sous-projet dans Polytech Génie électriques Projets (<http://forge.clermont-universite.fr/projects/polytechprojetsge>). Si le projet correspond à la suite d'un projet existant, utiliser de préférence le projet initial.

Par défaut, vérifier que le projet n'est pas public, ne rendre le projet Public que si la convention le permet.

Ajouter les différents utilisateurs à votre projet :

- étudiants,
- client,
- référent du projet,
- tuteurs extérieurs,

- tuteur industriel,
- responsable des projets.

Demander si nécessaire aux personnes de créer un compte en se connectant comme membre extérieur sur la forge.

4. Décomposition fonctionnelle du projet :

À partir du cahier des charges, le projet est découpé en fonctions à réaliser.

Pour chaque fonction il faut spécifier le rôle de la fonction (génération d'une haute tension...), définir les entrées et les sorties (que ce soit au niveau énergétique ou d'information), et créer les liaisons entre les fonctions permettant de réaliser le système complet. Ce schéma fonctionnel est déposé dans le wiki de la forge, sous forme d'image png et en fichier source joint, dans une page **[[schéma fonctionnel]]** au format yEd (<http://www.yworks.com/products/yed>).

Pour chacune des fonctions, vous devez spécifier la solution technique retenue en fonction des contraintes du projet et des ressources disponibles. Ceci vous conduira ensuite à créer le schéma structurel du système étudié.

Pour chaque fonction, avant de commencer le développement, vous listerez les éléments suivants :

- composants à provisionner et fournisseur,
- matériel de mise au point nécessaire,
- outils logiciels,
- ressources documentaires disponibles
- ... (si vous pensez à autre chose).

Cette liste devra être validée par le référent du projet au préalable à tout développement. Vous devez vous assurer que ces éléments seront accessibles lors de sénaces.

5. Intégration à la forge

L'organisation du travail se fera au travers des demandes (features (fonctionnalités), bug (erreur à corriger), support (demande d'assistance), task (tâche)).

Le découpage en fonctionnalités est calqué sur le découpage fonctionnel.

Pour chaque fonctionnalité, vous devez estimer le temps en heures nécessaire à sa réalisation.

Définir une fonctionnalité (nouvelle demande)

- Type : Feature
 - Partie ou sous partie du projet qu'il est possible de valider.
 - Associer la demande à une version cible (jalon), correspondant à la date où cette fonctionnalité doit être opérationnelle.
 - Conserver la date actuelle comme date de début, la chronologie des tâches sera mise en place par la suite.

- Définition de la date de fin :
 - La date de fin indique la durée prévue de la fonctionnalité en jours. Cependant, vous n'aurez pas projet tous les jours.
 - En fonction de la durée estimée de la fonctionnalité, vous pouvez vous reporter à cette table pour avoir la durée en jour en fonction de la période de l'année :

	Durée en jour en fonction du mois de réalisation							
Durée estimée en heures	Février	Mars	Avril	Septembre	Octobre	Novembre	Décembre	Janvier
2h	7	2	1	3	1	1	1	1
4h	14	4	3	7	2	2	2	2
8h	28	7	7	14	7	7	7	7
16h	--	14	14	28	14	14	14	14
n heures (semaines)	(n/2)	(n/4)	(n/8)	(n/6)	(n/8)	(n/8)	(n/8)	(n/8)

- Il sera possible de modifier cette durée par la suite.
 - Dans le descriptif de la fonctionnalité détailler le plan de test et de validation.
 - Créer la fonctionnalité
- Mettre à jour la fonctionnalité en ajoutant éventuellement des sous tâches ou fonctionnalités dérivées.
 - Utiliser le même principe pour définir la durée des sous-tâches ou sous fonctionnalité.

La granularité du découpage doit être comprise entre 8h et 24h.

Chaque fonctionnalité, sous fonctionnalité, ou tâche est ensuite affectée à un étudiant qui en aura la charge et la responsabilité.

6. Chronologie des travaux et Gantt

Après avoir créé les fonctionnalités et tâches, vous pouvez en organiser la chronologie. Pour cela, vous pouvez définir des relations d'ordre chronologique entre les tâches et fonctionnalités sous forme de liens.

Lorsque deux tâches ou fonctionnalités sont dépendantes (l'une ne peut être commencée qu'après la fin de la précédente).

Sélectionner l'une des deux demandes, ajouter une demande liée, choisir **suit** ou **précède** puis définir la tâche associée.

Lors de l'ajout d'une liaison, le Gantt est automatiquement mis à jour, la date de début de la tâche est alors automatiquement modifiée en conséquence. Vous pouvez intervenir sur la durée de la tâche en modifiant la date de fin (par rapport à la nouvelle date de début). Attention si vous imposez une nouvelle date de début celle-ci peut se retrouver en conflit avec les liaisons déjà établies.

Si deux tâches peuvent évoluer en parallèle tout en ayant une relation de dépendance créez une liaison de type **bloque**.

Associer ensuite les différentes demandes aux différents jalons (version) afin d'améliorer la visibilité de votre Gantt.

Chaque étudiant doit être affecté à au moins une tâche ou fonctionnalité pour chaque séance de projet.

7. Suivi du développement

À chaque séance de travail (pendant les heures de projet ou en dehors de celles-ci), veuillez mettre à jour le temps passé sur la ou les tâches ou fonctionnalités sur lesquelles vous avez travaillé, et mettez à jour l'avancement en % de la tâche. Ajoutez un petit commentaire décrivant le travail effectué et mettez à jour le temps passé (soit en l'intégrant à la mise à jour de la demande, ou en utilisant saisir temps).

Toutes les séances de travail, doivent être suivies de dépôt de nouvelles versions du code, schéma structurel, script, fichier texte contenant les développements mathématiques, ou autre sur le svn du projet. Lors de travail sur les présentations (qui doivent être fait en dehors des heures de projet), exporter la présentation dans son état actuel au format HTML ou texte et versionner le répertoire d'export.

Attention à ne jamais versionner des fichiers dont la taille dépasse 512Ko.

8. Utilisation de la forge lors des revues de projet

Le Gantt projeté lors des revues sera directement issu de la forge.

Pensez à projeter le gant complet (13 mois à partir de février de l'année en cours), puis le gant spécifique au jalon, vous argumenterez alors sur les tâches présentant un retard. Attention à l'échelle de temps.

Présentez les résultats obtenus au jalon actuel (demandes qui ont été fermées). Présenter explicitement les validations effectuées.

Pensez à présenter et lister de façon explicite sur un transparent les résultats qui seront présentés lors du prochain jalon.

9. Différents types de demandes

a. Fonctionnalité

Une fonctionnalité est un sous ensemble de votre projet pour lequel une validation est possible.

Une fonctionnalité peut être décomposée en plusieurs sous-fonctionnalités.

b. Tâches

Une tâche est un travail nécessaire pour réaliser une fonctionnalité. En général une fonctionnalité complexe est décomposée en plusieurs tâches.

c. Demandes de remontée d'erreur (bug)

Est soumis par un membre de l'équipe de projet et signale un dysfonctionnement à corriger rapidement. Elles peuvent aussi être soumises par vous même comme pense-bête afin d'éviter qu'un dysfonctionnement détecté ne soit oublié.

Ces demandes peuvent également être faites par les enseignants, par les clients.

Chronologie :

- La création de la demande se fait dans l'onglet nouvelle demande → « new ».

- Dès le début du travail sur cette demande, basculer la demande en « in progress ».
- Si la demande s'avère être inconsistante la basculer en rejetée « rejected ».
- Dès que la solution est trouvée passer la demande en résolue « resolved »,
 - le programme n'est peut-être pas encore utilisable.
- Dès qu'il est possible de tester la correction, basculer la demande à « feedback » et l'assigner à la personne qui a signalé le problème.
- Seule cette personne après tests peut fermer la demande → « closed ».

d. Demandes de support :

Il s'agit de demande d'aide que vous pouvez assigner à n'importe quelle personne (autres étudiant, ou équipe d'encadrement, personne extérieure).

Toute demande de support doit être définie à l'état nouveau, « new ». Vous renseignerez obligatoirement la durée estimée de traitement de la demande de support et l'affecterez à l'enseignant référent de votre projet. L'étudiant ou l'enseignant emmené à traiter la demande de support sera renseigné dans la première ligne de commentaire de la demande. La justification de la demande doit être la plus pertinente possible et fera partie des commentaires de la demande.

- Le référent du projet que vous aurez affecté en première intention validera la demande en la définissant comme « en cours » (« in progress »). Il l'affectera à l'étudiant désigné dans le commentaire de la demande.
- Il peut la rejeter « rejected » si il estime qu'elle fait partie des problématiques que vous devez être à même de traiter.
- Le nom de la personne qui doit traiter la demande sera défini dans le commentaire de la demande, vous aurez vérifié qu'elle apparaît bien comme membre de votre projet.

Vous surveillerez ensuite l'évolution de la demande, une fois qu'elle bascule en « feedback » il vous appartient de vérifier que le développement correspond à vos attentes et dès lors vous pourrez clore la demande.

Dans le cas où une demande de support est à destination d'un autre étudiant, la demande ne sera acceptée que si le projet de l'étudiant concerné ne présente pas de retard. Les référents des deux projets se concerteront et valideront la demande de support.

La sous-traitance initiée en début de 5a sera traitée comme une demande de support. La gestion correcte de ces actions de sous-traitance sera encouragée et appréciée lors de la notation du projet.

10. Utilisation du dépôt

Le dépôt sera de type svn.

Lors de chaque séance, une ou plusieurs remontées de données seront effectuées.

Les seules données versionnées sont les fichiers sources et les fichiers de configuration absolument nécessaire. Le non-respect de cette contrainte entraînant une surcharge inutile de la forge, elle aura une influence sur l'évaluation.

Une bonne pratique est de placer un nombre minimal de fichiers sur le dépôt lors de l'import ou de l'ajout initial. Toujours nettoyer le projet avant sa mise en place sur le serveur.

Cette partie du dépôt est ensuite idéalement descendue (checkout) sur une autre machine. Les fichiers manquants sont alors ajoutés sur la machine source, l'opération continue jusqu'à obtenir une configuration stable sur la machine originale et la machine de test.

a. Création du dépôt initial

Faire un checkout du dépôt initial dans un répertoire local `projets_ge_dev`, l'adresse apparaît dans l'onglet dépôt de votre projet.

Créer les sous-répertoires `trunk`, `branch`, `tags`

Dans le répertoire `trunk` créer les sous répertoires (à adapter en fonction de votre projet) :

- **CAO**
 - `carte`
 - `mécanique`
- **programmes**
 - `Microchip`
 - `nom_fonctionnalité` : un sous-répertoire par programme développé,
 - **Renesas**
 - `nom_fonctionnalité` : un sous répertoire par programme développé,
 - **Windows**
 - `langage_nom_fonctionnalité` : un sous répertoire par programme développé,
 - **Linux**
 - `langage_nom_fonctionnalité` : un sous-répertoire par programme développé,
- **présentations**
 - `export_html`
 - `yed` (fichiers source décomposition fonctionnelle)
- ...

Dans chaque répertoire, ajoutez un fichier `README.TXT` vide.

Une fois l'arborescence prête l'ajouter au dépôt de votre projet (faire `add` puis `commit`).

b. Mise en place d'un nouvel élément (programme, carte) :

Lors d'un nouveau développement :

- créer un répertoire correspondant au développement dans l'arborescence `svn`
 - créer un sous-répertoire si nécessaire,
- créer un projet minimaliste
 - Faire un « clean » du répertoire source avec l'outil de développement
 - Ajouter les fichiers à gestionnaire de version (`svn add`)
 - Rechercher dans la documentation de l'outil quels sont les fichiers à versionner.
 - Ajouter uniquement les fichiers nécessaires
 - pas de fichiers objets, pas de fichiers binaires ...
 - Envoyer les fichiers sur le serveur (`commit`)
- Afin d'éviter l'ajout malencontreux de fichiers générés, demander à `svn` de les ignorer (`svn ignore`)

Valider régulièrement que votre dépôt est correct en utilisation la procédure décrite chapitre 16 (Validation des dépôts).

11. Séances de travail :

À chaque fin de séance, veuillez livrer (commiter) les modifications, avec un commentaire explicite. Si possible ne livrer que des codes testables (c'est à dire qui compilent sans erreur). Si ce n'est pas possible, signaler lors de demande d'aide sur quelle révision les essais doivent être faits.

Mettre à jour le temps passé sur la tâche et l'avancement.

12. Documentations techniques

Les documentations techniques sont à charger dans la partie « documents » du projet de la forge. Attention, tenez à jour cette rubrique, elle sera utile lors de demandes aux enseignants, ils pourront récupérer les données des circuits sur lesquels vous travaillez rapidement.

13. Fichiers divers

Pour mettre à disposition différents types de documents ne devant pas être versionnés, utiliser les onglets fichiers et documents.

Les données de tests sont chargées sous forme d'archive zip dans la partie fichiers.

14. Maintenance

Pensez à vérifier et nettoyer les différents documents et fichiers mis à disposition régulièrement et à minima fin ge4a et fin ge5a.

Renseigner le contenu des différents fichiers README.TXT en décrivant les données disponibles dans les différents répertoires.

15. Première séance

- Créer une feature « cahier des charges », l'associer au jalon « revue préparatoire »,
- Créer une feature « découpage fonctionnel », l'associer au jalon « revue préparatoire »
- Créer une page wiki [[Cahier des Charges]], la compléter si possible ,
- Créer une page wiki [[Decoupage fonctionnel]], créer un schéma fonctionnel minimal (une seule boîte représentant le projet),
- Créer une feature « ajout des intervenants projets »,
 - contacter les différents intervenants à votre projet (client, tuteurs techniques ...)
 - leur demander de se connecter à la forge,
- créer le dépôt initial (voir 10.a),

Mises à jour, lors des séances suivantes :

- Dès que vous avez ajouté les différents intervenants en tant que membres de votre projet vous pouvez fermer la demande.
- En fin de séance, en fonction de votre travail saisir le temps passé sur les tâches correspondantes.

16. Validation de la cohérence du dépôt

Penser à faire ces opérations en fin de projet Ge4a et fin de projet Ge5a.

À l'issue du projet, les enseignants, clients, référents auront à intervenir sur vos projets. Cette étape sera de fait implicitement exécutée sur une nouvelle machine, il vous faut donc l'anticiper.

La validation du dépôt est une étape cruciale et complexe.

Vos projets peuvent fonctionner sur votre machine, tout en étant incomplets, pour les raisons suivantes :

- Certains fichiers non versionnés ou ignorés sont cependant présents sur votre disque, le projet fonctionne correctement,
- Des bibliothèques spécifiques, outils, compilateurs sont installés sur votre machine, garantissant le fonctionnement du projet.

Afin de garantir l'environnement de développement, il est nécessaire de le valider :

- Une machine vierge (VirtualBox) → pour une validation complète,
- La machine d'un collègue → la validation est partielle,
- Votre machine → validation très partielle.

Pour valider votre dépôt et l'environnement de développement :

- Faire un checkout du dépôt dans un répertoire projet_ge_production.
- Lister dans le README.TXT la liste des outils, version et lien de téléchargement utiles pour utiliser le projet.
- Vérifier que le projet est fonctionnel,
 - Ouverture du projet,
 - Compilation, génération de fichiers ...
 - Si nécessaire ajouter les dépendances à installer dans le fichier README.TXT
 - Si nécessaire ajouter les fichiers ignorés ou manquants à partir de répertoire de développement.
- Faire un commit des modifications des fichiers README.TXT