

Les projets font partie des modalités pédagogiques inscrites dans les programmes de première et de terminale NSI. Ce document a pour principal objectif d'aider les professeurs dans l'évaluation des projets en NSI. Le volume horaire et les contenus des programmes incitent à mettre en place durant l'année scolaire quelques projets et plusieurs mini-projets.

Au travers de la démarche de projet, les compétences à acquérir sont : Analyser, S'organiser, Programmer (Modéliser, Abstraire, Généraliser, Décomposer, Évaluer, Anticiper) et Communiquer.

## Cas des mini-projets

Un mini-projet dure moins de 8 heures et les groupes sont constitués de 2 ou 3 élèves.

On peut évaluer les élèves à l'aide d'un compte-rendu individuel, du code source et d'un oral.

Nous vous proposons une liste de critères d'évaluation qui devront être explicités aux élèves. Ces critères doivent être adaptés au type de projet. Il n'est pas nécessaire, ni forcément pertinent de tout évaluer à chaque fois.

### Pour le compte-rendu :

Proposition d'un guide de questionnement pour les élèves. Un exemple figure en annexe.

- Rappelez les éléments essentiels du cahier des charges.
- Décrivez la décomposition des tâches au sein de votre binôme et la méthode de travail.

Qui fait quoi ? Organisation temporelle (planning). Moments d'échanges et outils utilisés pour échanger.

- Présentez un sous-problème de vos tâches et la solution algorithmique choisie.

Exemple : pour trier une liste avec la méthode sélection, il faut d'abord trouver le minimum de la liste puis permuter ce minimum avec le premier élément de la liste ... : minimum et permutation sont des sous problèmes. La recherche du minimum utilise le parcours de liste déjà rencontré en cours, la permutation consiste en une affectation de valeurs dans un tableau, le tri par sélection est le programme principal utilisant les deux autres fonctions.

- Décrivez les spécifications liées à votre tâche.

Rôles, entrées, sorties, préconditions, postconditions, demande d'une preuve,

- Présentez l'interaction (IHM) que vous avez programmée ou un extrait du code suivant la nature du projet.
- Spécifiez les codes sources réutilisés dans vos tâches.

Quel code source (issu des TP, ou code trouvé sur internet) vous avez réutilisé ?

Quel processus de mise au point et de validation de programmes avez-vous utilisé ?

- Développez des processus de mise au point et de validation de tâches.

Quel processus de mise au point et de validation de programmes avez-vous utilisé ? Quels outils et méthodes de débogage ? Demande d'un jeu de tests.

- Pour aller plus loin : Proposer des améliorations possibles.

Au travers du compte-rendu, les critères d'évaluation pourraient être :

- l'organisation ;
- l'analyse des problèmes à résoudre ;
- la clarté de l'expression et des illustrations ;
- l'explicitation des méthodes et des choix effectués ;
- l'identification et le réinvestissement de notions vues en cours ;
- l'analyse critique des solutions.

## Pour le code source :

Au travers du code source, les critères d'évaluation pourraient être :

- le respect du cahier des charges ;

pour la partie programmation :

- la pertinence de la décomposition de la tâche sous forme de sous-tâches (fonctions, modules, bibliothèques, variables ...);
- la facilité de lecture du code : existence d'une documentation, de commentaires, de noms explicites pour les variables ;
- l'interopérabilité des programmes réalisés par chaque élève.

pour la partie interface graphique :

- la qualité du produit final (clarté, intuitivité, contenu, ergonomie...).

## Pour l'oral :

On peut ne demander une présentation orale qu'à un ou deux groupes par mini-projet. On pourra s'appuyer sur les compétences explicitées dans la grille d'évaluation du grand oral (note de service du 11 février 2020) et considérer qu'un niveau satisfaisant en classe de première ou début de terminale suffit pour avoir la note maximale.

On peut aussi inciter les élèves à faire une captation vidéo pour s'entraîner. Cette vidéo peut être évaluée à la place de la prestation orale en classe.

## Cas des projets

Les projets sont mis en œuvre sur un temps long (au moins 3 semaines).

On peut évaluer les élèves à partir d'un compte-rendu individuel, de revues de projet, du code source et d'un oral collectif. Les critères d'évaluation sont les mêmes que ceux définis pour les mini-projets.

Les objectifs du projet sont décrits par un cahier des charges détaillé et fourni par le professeur. Ce cahier des charges pourra être réajusté ou affiné, par l'équipe de projet en fonction de leur analyse (ouverture d'esprit, implication des élèves, ...).

Dans la gestion de projet, il est important de planifier des moments d'échanges au sein de l'équipe, appelés revues de projet, dont le nombre dépend de la taille et de la complexité des projets.

## Première revue de projet :

Les objectifs de cette revue de projet sont :

- Analyser le besoin.
- Rechercher des premières pistes de solution (découpage du problème).
- S'organiser :
  - partage des tâches ;
  - amorce de planification.

Cette première revue de projet se fait à la fin de la première semaine ou au début de la seconde semaine avec l'équipe au complet et ne devrait pas donner lieu à une note. On s'assure ainsi que le groupe a effectivement compris le problème à résoudre et qu'il a mis en place une organisation adéquate.

À l'issue de cette revue de projet, l'équipe doit rédiger un relevé de conclusion. L'existence et la pertinence de ce document peut entrer dans l'évaluation du projet.

## Revue de projet suivantes :

Durant le projet, chaque élève effectue sa tâche en autonomie avec l'aide ponctuelle du professeur. Le code qu'il crée doit pouvoir s'intégrer dans le projet global de l'équipe. Des moments de rencontres entre les membres de l'équipe, appelés revues de projet, doivent donc avoir été planifiés avec ou sans le professeur.

Chaque début et fin des séances doivent être consacrés à un moment, même court, durant lequel l'équipe identifie le travail fait et celui restant à réaliser. L'équipe peut utiliser un cahier de suivi résumant les bilans de ces moments.

À la fin du projet, l'équipe doit s'assurer de l'interopérabilité des différentes briques (par exemple : les noms de variables ou de fonctions sont-ils cohérents ? partagés par tous ?).

## Présentation orale du projet

Cette présentation se veut collective. Pour préparer cet oral, chaque élève doit rédiger un compte-rendu sur le modèle proposé pour les mini-projets. Il doit être capable d'expliquer comment s'intègre sa tâche dans le projet global et critiquer le résultat obtenu. Une partie du compte-rendu peut être commune à tous les membres de l'équipe.

L'exposé oral peut commencer par une explicitation des conclusions de la première revue de projet et se poursuivre par les présentations individuelles du travail de chacun.

Annexe : Extrait d'un mini-projet proposé par S. Meal, professeur au lycée J. Moulin à Angers

|   |                          |  |
|---|--------------------------|--|
| Numérique et Sciences Informatiques   |                          |  |
| 6h  | <b>Projet : site Web</b> |  |
| Objectif : écrire du HTML, CSS et JS.   |                          |  |
| <b>Matériel</b> : Notepad++ ou repl.it pour travailler sur le même code à plusieurs |                          |  |



## Conditions

Le travail est à réaliser en binôme à la maison (vous pouvez utiliser **repl.it**), les tâches devront être réparties équitablement.

Des échanges entre les membres du binôme devront se faire régulièrement pour que le projet aboutisse.

|                 |  |  |
|-----------------|--|--|
| <b>Groupe 1</b> |  |  |
| ...             |  |  |

## Cahier des charges

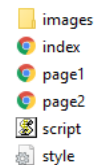
Vous allez réaliser un site sur le sujet de votre choix (nous pourrions mettre votre site en ligne si vous souhaitez donner une visibilité à votre travail).

Vous ne serez pas évalué sur le contenu ni la forme de votre site (à moins qu'il n'y ait pas de travail) mais sur un rapport à produire individuellement et le respect des éléments suivants :

- ❖ Le site doit respecter la structure vue en cours avec les éléments principaux (header, main, menu de navigation, footer et si besoin un aside). La mise en forme (le placement des éléments et leur style) ne doit pas être la même que celle proposée en cours (Soyez originaux !)
- ❖ Le site doit posséder plusieurs pages et donner la possibilité de naviguer entre ces différentes pages.
- ❖ Le site doit posséder au moins un tableau.
- ❖ Le site doit posséder un formulaire qui permet de récupérer des valeurs dans une fonction JS. Les valeurs de chaque champ seront traitées et le résultat de ce traitement apparaîtra dans votre site. Ex : on rentre des notes et on affiche la moyenne.
- ❖ Le site doit proposer deux interaction (une par élève) avec l'utilisateur (autres que hyperlien et formulaire).

Ex : l'appui sur bouton qui déclenche une fonction JS.

- ❖ Structure du site (nommez page1, page2 ... comme vous le souhaitez :



Ne cherchez pas à faire un site avec du code que vous ne comprenez pas, un site simple fera l'affaire.

## Les parties du rapport individuel (un rapport par élève)

Partie 1: Décrivez la décomposition des tâches au sein de votre binôme et la méthode de travail.

Qui fait quoi ? Organisation temporelle (planning). Moments d'échanges et outils utilisés pour échanger.

Partie 2 : Présentez un sous-problème de vos tâches en les rattachant à une situation déjà rencontrée en cours, et la solution choisie.

Ex : pour trier une liste avec la méthode sélection, il faut d'abord trouver le minimum de la liste puis permuter ce minimum avec le premier élément de la liste ... : minimum et permutation sont des sous problèmes. La recherche du minimum utilise le parcours de liste déjà rencontré en cours, la permutation consiste en une affectation de valeurs dans un tableau, le tri par sélection est le programme principal utilisant les deux autres fonctions.

Partie 3 : Présentez l'interaction que vous avez programmé.

Expliquez quels éléments de votre site permettent une interaction avec l'utilisateur et comment ?

Partie 4 : Spécifiez les codes sources réutilisés dans vos tâches.

Quel code source (issu des TP, ou code trouvé sur internet) vous avez réutilisé ?

Quel processus de mise au point et de validation de programmes avez-vous utilisé ?

Partie 5: Développez des processus de mise au point et de validation de tâches.

Quel processus de mise au point et de validation de programmes avez-vous utilisé ? Quels outils et méthodes de débogage ?

Partie 6: Pour aller plus loin : imaginez que votre site soit en ligne, quel service souhaiteriez-vous ajouter nécessitant l'utilisation d'un langage de programmation côté serveur (style PHP) et dont l'utilisation seule de HTML+CSS+JS n'est pas suffisante.