

1 Le cahier des charges

1. Les groupes de 2 ou 3 sont imposés et les sujets sont déjà répartis dans le fichier suivant :

<https://nuage03.apps.education.fr/index.php/s/PyZoyRPk2rFNCGb>

2. Le rendu doit s'effectuer sous trois formes, au plus tard le **jeudi 15/05** :

- **Mardi 28/04/2026** : un **premier point étape** collectif rendu sous la forme d'un entretien oral avec l'enseignant sur **1 point** et d'une fiche de suivi de projet sur **1 point**. Lien vers la fiche de suivi de projet :

<https://nuage03.apps.education.fr/index.php/s/sFHfRL8kdi2tQaf>

- **Mercredi 13/05/2026** : un **rendu collectif** (un seul par groupe de projet) :
 - **Sur 3 points** : le code écrit dans le cadre du projet : sont évalués sa correction (exécution sans bug), sa qualité (richesses et efficacité des structures de données ou de contrôle, modularité), sa lisibilité (noms de variables, commentaires).

- **Mercredi 20/05/2026** : un **rendu individuel** :

- **Sur 5 points** : sous la forme d'une capsule vidéo de 4 minutes minimum à 8 minutes maximum au format mkv ou mp4. Vous pouvez utiliser Vokoscreen : <https://linuxecke.volkoh.de/vokoscreen/vokoscreen-download.html>.

La capsule doit être structurée en au moins 4 parties :

- * Une introduction avec présentation des objectifs du projet.
- * Une partie sur l'organisation du travail au sein du groupe (répartition des tâches, planification)
- * Une partie expliquant un ou plusieurs extraits du code
- * Une conclusion sur ce que vous avez appris ou développé comme compétences dans ce projet.

2 Quelques règles à respecter

- **Règle n°1** Avant de coder je réfléchis crayon en main à l'architecture de mon programme et j'essaie de bien distinguer les données des fonctionnalités..
- **Règle n°2** J'applique le principe de programmation modulaire :« *Chaque fois qu'une tâche peut être clairement séparée dans un programme, je l'encapsule dans une fonction et chaque fonction peut être écrite par une personne distincte.* »..
- **Règle n°3** J'écris des tests unitaires ou j'exécute les tests fournis pour chaque fonction de mon programme.
- **Règle n°4** Je prête attention à la lisibilité de mon programme : noms de variables et de fonctions explicites, autodocumentation par une docstring pour chaque fonction et commentaires pertinents des points les moins lisibles.
- **Règle n°5** Je transmets au professeur un Notebook Capytale ou un script Python dans un fichier texte d'extension .py.
- **Règle n°6** Je ne perds jamais de vue mes objectifs, je fais un point régulier avec les membres de mon trinôme et les professeurs , j'utilise des outils de travail collaboratif.