

# De l'art du peer-correcting – intraV2

---

Ce petit recueil cherche à décrire l'état d'esprit avec lequel il semble préférable et intéressant que vous abordiez toute soutenance, ainsi que quelques obligations pour ceux qui ne saisiraient pas correctement cet état d'esprit.

42 c'est une communauté. Vous faites partie d'un tout. De la même façon qu'il est légitime que vous puissiez contribuer à l'ambiance, à l'avancement de chacun, aux ressources disponibles (peut-être même un jour, plus tard, à certaines évolutions pédagogiques sur la forme ou le fond), il est naturel que vous contribuiez à l'évaluation de tous les étudiants.

Comme vous discutez tout le temps avec plein de gens, que vous échangez, que vous montrez votre code, vous avez en permanence un retour direct de la communauté sur ce que vaut votre production. Vous voyez bien si ça marche ou pas, et si vous aurez une plutôt bonne ou plutôt mauvaise note. Bref pas vraiment de surprise lors de la soutenance. La note est surtout là pour que le staff puisse suivre l'évolution de chacun, et éviter que certains ne se cachent à eux-mêmes la réalité de leur travail (si-si ça existe).

Si vous avez été attentif aux vidéos et documents sur la pédagogie du début de piscine, il y est fait mention d'un processus d'avancement dans les projets, avec lecture/débat sur le sujet, puis sur les pistes à suivre, les tests, re-débat, ainsi de suite jusqu'à l'aboutissement. Or ce processus contient également une dernière étape d'assimilation, de bilan, de retour à posteriori sur le projet et les difficultés, les stratégies adoptées, ce qui a fonctionné et ce qui a lamentablement échoué. Bref, une analyse avec du recul sur ce que vous avez fait. Alors que vous êtes déjà en train de courir sur les nouveaux projets, cette étape du bilan est souvent bâclée, voire mise de côté. Or c'est une des plus importantes pour votre propre apprentissage, quel que soit ce que vous avez pu produire. Chez 42, le seul moment propice à cela, c'est la soutenance, même si elle est là dans un premier temps pour l'évaluation. Ce moment est à exploiter à son maximum pour capitaliser vos expériences et en construire de meilleures par la suite.

La multiplicité des soutenances en peer-correcting est importante, pour à la fois un effet statistique, en éliminant les effets de bord (trop bien ou trop mal noté), et à la fois un effet psychologique, pour ne pas retomber dans les travers de la notation unique par un prof. Il est donc important de faire le plus possible de soutenances, et profiter de la totalité des interlocuteurs, tant comme noté que comme noteur. Ce n'est en aucun cas du temps perdu.

Pour nous, une soutenance c'est :

- Un contact. Un échange. Cela se fait en présentiel, tout le monde est là. Pas du mail ou du téléphone (ou seulement avant pour fixer le rendez-vous, mais même ça, aller voir l'autre en cluster, ça me paraît mieux). On git clone uniquement à ce moment-là.
- Prendre le temps. Le temps de s'arrêter, de dialoguer, de découvrir l'autre. Chaque interlocuteur est différent, chaque soutenance est unique. Quel que soit le rendu fait (ou pas

fait), une soutenance doit durer le temps prévu (comptez env. 15 à 20 min, un peu plus pour les gros projets).

- Si une norme s'applique sur le projet, elle est vérifiée en premier lieu, avec la norminette officielle dès lors que celle-ci est présente. Cette étape ne doit rien au hasard. En effet il ne s'agit pas pour vous d'être infaillible lorsque vous codez, mais de développer votre capacité de mise en œuvre d'éléments de vérification et de contrôle de votre travail. Si la norme ne va pas, l'évaluation avec le barème s'arrête.
- Constater, ensemble, ce qui fonctionne, et ce qui fonctionne pas. Etre tous d'accord sur le résultat des tests ainsi que sur l'application du barème et la note. Cette étape est avant tout fonctionnelle, tel le client qui vérifie que tous ses besoins sont remplis sans se demander si c'est de la table de hash ou de la liste chaînée (on ne regarde pas comment c'est fait, seulement si ça marche ou non).
- Bidirectionnel. Chacun évoque ses idées et hypothèses sur ses propres solutions, sur la pertinence et la qualité du produit, ses facteurs de réussite ou d'échec, comme ceux des autres. C'est comme un dialogue avec un collègue de travail.
- Constructif. Tout le monde repart avec quelque chose en plus, tant technique que relationnel ou organisationnel.

On aurait pu mettre dans cette liste tout un tas d'obligations et d'interdiction, mais je préfère que chacun s'approprie cet état d'esprit parce qu'il a du sens, pour chacun comme pour tous. Enfin, les feedback sont importants, utilisez-les pour témoigner de la qualité d'une soutenance.

Comme malheureusement mon paragraphe précédent risque pour certains de rester lettre morte, on va récapituler les choses de la façon suivante :

- Toute soutenance doit se faire **OBLIGATOIREMENT** en présentiel.
- Le nombre de notes reçues doit atteindre le seuil demande, faute de quoi les compétences ne sont pas attribuées en cas de réussite.
- Ce même nombre de notes s'applique aussi aux notes données, et doit logiquement être atteint, sinon vous perdrez votre capital de points de soutenance et ne pourrez plus être notés par la suite.
- On ne **REFUSE PAS** une soutenance qui a été organisée, ni comme noteur, ni comme noté.
- Le feedback est **OBLIGATOIRE**. Il se fait immédiatement en fin de soutenance. Tant que tous les feedback ne sont pas effectués, les skills ne sont pas attribués.

Ces éléments sont essentiels au bon fonctionnement du peer-correcting, et font partie du deal de l'école avec vous. Ne pas les respecter vous expose à des **SANCTIONS**, pouvant aller de l'invalidation d'une ou plusieurs notes, jusqu'à un arrêt de votre scolarité.

Encore une fois, ces éléments doivent être ressentis comme naturels dans la logique de votre scolarité, dans la logique communautaire de l'école. Si vous avez l'impression qu'il s'agit d'une contrainte, c'est que votre modèle de travail doit encore évoluer : capacité de travail, investissement temporel, organisation personnelle, relationnel et communication... .