

Le barème et les attentes d'une soutenance de projet

Dans la suite, la ou les personnes qui évalueront la soutenance seront appelées «le jury» et le, la ou les étudiants qui sont évalués seront appelés «le soutenant».

Le but principal de la soutenance de projet sert à évaluer la compréhension par le soutenant du code informatique du projet. Le but secondaire, et moindre, est de permettre au jury de pouvoir éclaircir certains points du projet.

Pour la soutenance, le soutenant doit ...

- ...se préparer à répondre aux questions qui porteront sur le but du projet, la "manière de l'utiliser" et surtout sur le code source du projet.
- ...préparer l'intégralité du code source du projet, de façon qu'il puisse être rapidement consultable, modifiable et expérimenté durant la soutenance, même si ce code est éclaté sur plusieurs fichiers (par exemple les fichiers sources peuvent être ouverts dans différents onglets d'un éditeur de texte graphique).
- ...signaler, en début de soutenance, la présence de parties du code qu'il n'a pas comprises et les avoir marquées, par exemple avec un commentaire.

On pourrait décrire ainsi le barème de la note projet+soutenance : Le projet est évalué une première fois, le jury pose des questions et consulte le code afin de bien cerner l'étendu des fonctionnalités du projet et des méthodes utilisées/écrites, ce qui donne une première note (non communiquée). Puis ensuite, dans une seconde phase, le jury évalue la compréhension du code par le soutenant (et donc implicitement la capacité du soutenant à l'expliquer). Enfin le jury décide d'une note finale.

La note finale peut être inférieure à la première note si le jury estime que le soutenant ne maîtrise pas une partie du code.

Le soutenant peut éventuellement commencer la soutenance par une présentation, bien que ce ne soit pas obligatoire. Celle-ci ne devrait pas durer plus d'un cinquième du temps totale de la soutenance. Elle pourra éventuellement être écourtée par le jury.

Idéalement, la présentation ou la forme (diction, esthétique) de la soutenance n'entre pas en compte dans la note.

Le soutenant doit convaincre le jury qu'il maîtrise le code produit. Il ne doit pas hésiter à expliquer des concepts (ou mécanismes ...) même si ceux-ci lui semblent évident. Le jury peut très bien prendre l'initiative d'écourter une explication (il ne faut pas s'en formaliser la raison est le temps qui est limité) ou au contraire de demander des détails mais par défaut le soutenant devrait faire comme si le jury était ignorant (ce qui n'est pas improbable) mais possède quelques bases en programmation.

Le soutenant doit être précis dans ses réponses. Une information fausse donnée par le soutenant (même une faute par paresse d'expression orale) pourrait être interprétée comme une méconnaissance et être pénalisante.

Il est possible d'utiliser du code (acquis par ailleurs) que le soutenant ne maîtrise pas, mais cela peut engendrer des pénalités. Ce code doit par ailleurs clairement être marqué comme non compris (par exemple avec du code de type commentaire). De plus, surtout, un tel code doit être signalé en début de soutenance, autrement cela pourrait être interprété comme une tentative de fraude (et entraîner de très grosses pénalités).

Des outils externes (comme par exemple des fonctions provenant de bibliothèques autorisées) utilisés dans le code source devraient idéalement être grossièrement maîtrisés par le soutenant. Ce dernier devrait pouvoir imaginer une manière de les remplacer par des outils qu'il aurait lui-même conçus (sans qu'il n'ait à rentrer dans des détails comme les appels systèmes/utilisation de matériel, l'organisation de code, contrainte de performances).