

NoteR

Les auteur.ices

Encadrante : Laurence Rozé

2024-2025

Résumé

TODO

Introduction

Notre encadrante, Laurence Roze, est, depuis l'an dernier, repsonsable des deuxièmes années au département Sciences et Techniques Pour l'Ingénieur (STPI) à l'INSA. Cette fonction l'amène à la gérer l'ensemble de la pormotion afin de veiller à son bon encadrement. Cela implique notamment la gestion des contrats de redoublement, la création des emplois du temps, la répartition des élèves par groupes, ... À sa prise de poste, Mme Rozé s'est rendue compte que, peu, voire aucunes de ces opérations n'étaient automatisée, ce qui demande donc un travail long et fastidieux, pouvant amener à des erreurs. Dans le cas des contrats de redoublements, les notes étaient écrites manuellement dans les contrats, ce qui pouvait amener à des erreurs de saisie. C'est dans ce contexte que Mme Roze nous a demandé de développer un outil capable d'automatiser le processus de création des contrats de redoublement. Ce dernier devait, à partir du fichier de jury, fichier récapitulatif des notes de tous les étudiants de la promotion, du fichier d'en-tête, regroupant les décisions du jury de fin d'année, et d'un fichier de constantes, descriptif de la maquette, générer un contrat vierge, le contrat de tous les étudiants redoublants et un fichier récapitulatif des matières suivies l'année d'après par chaque étudiant. Ce rapport détaillera les différentes étapes de la conception de notre outil, NoteR, ainsi que l'enjeu de chacune d'elles, depuis les outils utilisés jusqu'à la conception de l'outil en lui-même et des choix qui l'ont accompagné.

1 Outils utilisés

1.1 Langage de programmation

L'intégralité de NoteR est codée en R. Cette contrainte était imposée dans le sujet et ce choix peut être motivé par plusieurs raisons :

- NoteR doit s'intégrer dans un outil préexistant développé en R par Mme Rozé,
- R est un langage de programmation qui n'est pas aussi régulièrement mis à jour que d'autres langages, ce qui permet d'assurer la longévité de NoteR,
- R est un langage traditionnellmeent utilisé pour les statistiques et Mme Rozé voulait se laisser la possibilité dans le futur de calculer des statistiques sur les redoublants de deuxième année.

Nous avons ainsi utilisé plusieurs libraries de R et particulièrement, pour la conception de l'interface graphique de NoteR, la librairie RShiny.

1.2 Outils collaboratifs

Afin de faciliter la gestion du projet et notamment les mises à jour du code, nous avons choisi d'utiliser la plateforme GitHub, qui permet à tous les membres du projet d'avoir accès en temps réel à la dernière version du code mise en ligne. De plus, n'étant au début pas très à l'aise avec cet outil, nous avons préféré utiliser Google Drive pour tous les documents annexes comme les comptes-rendus de réunion ou le suivi des tâches.

2 Génération notes

2.1 Génération des notes

La première fonctionnalité développée dans le projet est la génération automatique de notes. Elle a joué un rôle essentiel pour tester l'ensemble des autres modules (contrats pédagogiques, bilans de notes, interface utilisateur) sans utiliser de données sensibles ou incomplètes.

Le script `generer_notes_automatique.R` permet d'attribuer à chaque étudiant trois évaluations (DS, CC, TP) par élément constitutif (EC), avec des notes aléatoires comprises entre 0 et 20, arrondies à deux décimales.

Une fois les notes générées, le script `calcul_moyennes.R` calcule : la moyenne de chaque EC, la moyenne de chaque UE et le statut de l'UE (**VALIDE** si la moyenne ≥ 10 , sinon **NON VALIDE**).

Pour adapter ces données au format attendu du fichier `jury.xlsx`, la fonction `creer_ligne_unique.R` extrait et organise les résultats par étudiant et par semestre (S3, S4) obtenus, dans une structure claire et unifiée.

Enfin, le script `write_data_to_sheet.R` exporte les résultats dans un fichier Excel `jury.xlsx`, avec une feuille dédiée à chaque semestre.

Cette base de données synthétique est utilisée par les autres modules : génération des contrats pour les redoublants, bilans individuels, et suivi global des validations.

2.2 Entête Jury

La fonctionnalité **Entête Jury** génère un fichier Excel listant les étudiants n'ayant pas validé leur année, en vue des délibérations pédagogiques.

Le script `validation_globale.R` sert à valider automatiquement un étudiant pour un semestre donné (S3 ou S4) en fonction des moyennes pondérées par ECTS dans différentes UE. Il retourne un tableau indiquant si chaque étudiant valide ou non son semestre, avec les moyennes associées. Un semestre est marqué comme non validé si au moins une UE ne l'est pas. Contrairement à l'étape précédente où le statut est calculé par UE, ici il s'agit d'un statut global par semestre.

Une fois cette validation effectuée pour tous les étudiants, le script `filtrer_non_valide.R` extrait ceux qui n'ont pas validé l'année, c'est-à-dire qui ont échoué à au moins un des deux semestres.

Enfin, `to_EnteteJury.R` exporte ces résultats dans un fichier Excel nommé `EnteteJury.xlsx`.

Pour la simulation, une décision finale est ensuite attribuée à chaque étudiant : **Passe**, **Red** (redouble), ou **Exclu**. Cette étape reste aléatoire à ce stade car d'autres critères doivent être pris en compte pour prendre la décision finale, mais elle permet de visualiser le format attendu par l'administration.

3 Génération contrats redoublants

La génération des contrats repose entièrement sur le fichier **MCC**. Toute modification du **MCC** (comme les noms d'UE ou d'EC) est automatiquement prise en compte sans avoir besoin de modifier les programmes de génération de contrats.

3.1 Contrat vierge

Le contrat vierge est un document Word généré automatiquement à partir des données contenues dans le fichier **MCC**. Il n'inclut pas encore les notes d'un étudiant, mais est entièrement adapté aux informations de l'année en cours (noms des UE, EC, etc.). Le code de génération n'a pas besoin d'être modifié chaque année : il s'adapte dynamiquement aux données du fichier **MCC**.

Voici un aperçu de l'en-tête du contrat vierge :

CONTRAT D'ÉTUDES 2 STPI

Nom et prénom de l'étudiante : _____

Semestre : ☐ 3 ☐ 4 Année complète : ☐

Motif de l'établissement du présent contrat :

☐ SHN ☐ Redoublement ☐ Redoublement de Cas de Force Majeure

☐ Autre (à préciser) : Dossier médical

Détails du contrat d'études :

UE	EC	Code EC	EC 2024-2025 (Moyenne obtenue)	EC validé mais repassé en 2025-2026	EC à valider en 2025-2026
Sciences expérimentales (UE-STP03-SE)	Systemes Automatisés	EC-STP03-ACSA			
	Chimie 3	EC-STP03-CHIM			
	Electronique 1	EC-STP03-ELEC			
	TP Physique 3	EC-STP03-PHYS			
	Thermo-énergétique	EC-STP03-THEN			

FIGURE 1 – En-tête du contrat vierge

Le contrat contient également un espace dédié à la signature, situé à la fin du document. Cet espace est prévu pour :

- la signature de l'élève précédée de la mention « lu et approuvé »
- la signature de la direction du département STPI
- l'apposition du cachet de l'établissement

L'élève s'engage à respecter le présent contrat signé.

Signature de l'étudiant·e précédée de la mention «lu et approuvé»	Signature de la Directrice du département : Carole Daigebonne	Cachet de l'établissement
Date :	Date :	

FIGURE 2 – Espace de signature du contrat

3.2 Contrat avec notes

Une fois les notes générées dans le fichier `jury.xlsx`, il est possible de générer automatiquement un contrat personnalisé pour chaque étudiant redoublant à partir de son identifiant.

La génération repose sur la fonction `generation(id, doc)`, où `id` désigne l'identifiant de l'étudiant. Cette fonction :

- extrait automatiquement le **nom** et le **prénom** de l'étudiant à partir de son identifiant
- insère ces informations en haut du contrat
- ajoute ses moyennes dans la colonne «**EC 2024-2025 (Moyenne obtenue)**»
- coche les EC dans la colonne «**EC à valider en 2024-2025** » si :
 - la moyenne de l'EC est strictement inférieure à 10 ou absente
 - et l'UE correspondante est indiquée comme **non validée**
- intègre un tableau de signature à la fin du contrat

4 Interface graphique

À mesure de l'avancée du projet et après avoir discuté avec Mme Rozé de l'utilisation future de NoteR, nous avons convenu de développer une interface graphique. En effet, même si ce n'était pas demandé dans les contraintes initiales, dans une démarche d'améliorations de l'outil pour une utilisation par un futur responsable d'année non-informaticien, nous pensons que c'est le meilleur moyen de le rendre intuitif et facile d'utilisation. Nous avons donc regroupé les différentes fonctions en 4 boutons : un pour générer le tableau contenant les notes de l'étudiant sur lequel on peut rajouter à la main les croix correspondant aux EC déjà validés mais qui seront repassés l'année suivante, un pour la génération du contrat de l'étudiant sélectionné, un pour la génération de l'ensemble des contrats des redoublants de deuxième année et enfin un pour la génération du fichier récapitulatif. Cette interface ne prend en entrée que le fichier d'en-tête, afin de récupérer la liste des étudiants redoublants et le fichier de jury, nécessaire à la création du tableau de notes de l'étudiant courant. L'interface permet donc de générer facilement l'ensemble des sorties attendues et minimise le risque d'erreur humaine.

(Image de l'interface à insérer quand elle sera finie)

5 Génération du bilan

5.1 Filtrage des étudiants redoublants et génération des contrats

Dans le cadre de ce projet, nous avons développé un ensemble de scripts R visant à automatiser le processus de génération des contrats de redoublement à partir des données du jury. Ce processus se déroule en plusieurs étapes distinctes.

5.1.1 Filtrage des étudiants redoublants

La première étape consiste à écrire un script nommé `filtrer.R`. Ce script permet d'identifier les étudiants susceptibles de redoubler, en se basant sur les données issues du fichier `entete_jury.csv`. Il est important de noter que les simples moyennes présentes dans le fichier `jury.xlsx` ne suffisent pas à déterminer les cas de redoublement ; d'autres critères doivent être pris en compte, d'où l'utilisation de ce filtrage préalable.

5.1.2 Génération des contrats

Une fois les étudiants redoublants identifiés, nous utilisons un second script appelé `listeContratAlgo.R`. Ce dernier automatise l'appel à la fonction `generation()` définie dans le fichier `contrat_notes.R`. Cette boucle permet de générer individuellement un contrat de redoublement pour chaque étudiant concerné, en utilisant les données du fichier `jury`.

5.1.3 Production des bilans

Enfin, une fois l'ensemble des contrats généré, nous produisons des bilans synthétiques à l'aide d'un script final. Ces bilans sont générés pour :

- le semestre 3 (S3),
- le semestre 4 (S4),
- et l'ensemble de l'année.

Ces bilans contiennent la liste des étudiants redoublants ainsi que les unités d'enseignement qu'ils devront repasser lors de la prochaine année universitaire. Ils constituent un outil précieux pour l'administration pédagogique dans le suivi des parcours étudiants.

6 Bilan

Nous avons donc réalisé le travail demandé :

- Générer un contrat vierge
- Générer un contrat avec des notes
- Générer l'ensemble des contrats des redoublants avec leurs notes
- Générer un bilan

De plus, nous avons réalisé une interface graphique permettant :

- De générer le contrat d'un redoublant en cliquant sur le bouton « Génération du contrat de l'étudiant sélectionné »
- De générer le fichier bilan en cliquant sur le bouton ...
- De générer l'ensemble des contrats en cliquant sur le bouton « Génération de l'ensemble des contrats des redoublants »

- D'ajouter des croix dans la colonne « EC validé mais repassé en 2025-2026 » du contrat (ceci est un ajout au cahier des charges initial, il a été proposé par notre groupe afin de faciliter l'utilisation de NoteR)

Point important : d'une année à l'autre la maquette peut évoluer. Elle est décrite dans un fichier d'entrée MCC permettant à notre code de s'adapter automatiquement à celle-ci.

Conclusion

Notre étude pratique est terminée et propose un outil complet permettant de générer les contrats des redoublants. Il va au-delà du cahier des charges initial, proposant en plus une interface graphique permettant de sélectionner les matières déjà validées, mais que l'élève peut choisir de repasser pour améliorer sa note.

Cependant, l'outil que nous avons développé reste une première version. Il n'est pas accessible à n'importe quel utilisateur sans connaissances techniques, notamment en ce qui concerne la modification du fichier MCC, qui nécessite des bases en programmation. Une amélioration importante consisterait à rendre l'outil plus accessible à des utilisateurs non experts.

Ce projet annuel a représenté un véritable défi. Il nous a permis de nous confronter à un développement concret sur le long terme, de renforcer notre capacité à collaborer efficacement, à nous organiser et à faire évoluer un projet en fonction de retours et de contraintes réelles.