



HAL
open science

PLaTon, une plateforme pour apprendre et enseigner

Dominique Revuz

► **To cite this version:**

Dominique Revuz. PLaTon, une plateforme pour apprendre et enseigner. LIGM - Laboratoire d'Informatique Gaspard-Monge. 2020. hal-03920164

HAL Id: hal-03920164

<https://univ-eiffel.hal.science/hal-03920164>

Submitted on 3 Jan 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

PLaTon, une plateforme pour apprendre et enseigner

PLaTon : Platform for Learning and Teaching online

Dominique Revuz, Univ Gustave Eiffel,
Laboratoire Informatique Gaspard Monge Univ Gustave Eiffel,
CNRS, LIGM, F-77454 Marne-la-Vallée, France

Abstract: L'objectif de cet article est de présenter les principes fondateurs de la plateforme d'enseignement PLaTon et d'exposer les principales raisons qui ont motivé son développement. Un contexte sociétal et historique, l'augmentation du nombre de bacheliers, la pandémie covid, la position de la France dans les classements sur l'enseignement des mathématiques. Les solutions existantes: des exercices, WIMS, LMS, quiz, ASKER, COMPER, etc. Les éléments d'architecture, capacité de monter en charge, sécurité, RGPD, ouverture à d'autres formats. Les besoins visés: partage, accessibilité, gratuité, ergonomie, capacité d'évolution et d'innovation, indicateurs et tableaux de bords, exercices auto-corrigés répétables, activités didactiques, gestion de la métacognition, pluri-disciplinarité, adaptativité, analytics.

1. PLaTon la construction du projet.

PLaTon est une plateforme de type "software as a service" (saas) qui fournit un ensemble d'outils au service de l'enseignement. Notre ambition est de produire un outil communautaire ouvert capable de répondre aux besoins des élèves, des enseignants et des chercheurs tout en répondant au besoin naturel de la société d'améliorer son enseignement à tous les niveaux et à tous les âges.

PLaTon est un projet qui a pris de l'ampleur avec le temps et qui est conçu pour pouvoir encore évoluer dans ces missions et ces objectifs. Initialement, ce qui était visé était un exercice de programmation avec une syntaxe la plus simple possible et ne nécessitant qu'une connaissance basique de la programmation Python. L'exercice devait être résistant aux attaques (les élèves pouvant fournir éventuellement à leur insu des bombes logicielles comme réponses, en particulier des utilisations excessives de ressources système). Une fois cette étape franchie avec le logiciel PremierLangage (prémisse de PLaTon). Le projet a rencontré d'autres disciplines (maths, chimie, physique, lettre, économie, gestion, anglais) avec leurs nouveaux besoins. La nécessité d'avoir une réflexion plus large et plus profonde s'est avérée nécessaire.

Le contexte sociétal et historique a aussi été une forte motivation pour redéfinir les ambitions de notre équipe. La réorganisation de l'enseignement secondaire, réalisée depuis des décennies, avec son impact sur le nombre de bacheliers sont autant d'enjeux que l'université doit maintenant relever. La capacité à apprendre des jeunes étudiants a changé les modalités d'enseignement des premières années à l'université. La diminution mécanique du ratio nombre d'enseignants sur nombre d'étudiants, avec l'augmentation du nombre d'étudiants non compensée par une augmentation du nombre d'enseignants dans le supérieur. La pandémie covid qui a montré l'importance des outils en ligne dans l'enseignement, et enfin les résultats PISA ou la position de la France dans les classements sur les résultats de l'enseignement n'est pas digne de notre histoire.

L'équipe s'est donc lancée dans une conception plus formelle du projet avec une ouverture plus large sur les services que doit rendre une plateforme d'enseignement en ligne.

Les parties Prenantes

Les parties prenantes principales sur PLaTon sont les apprenants, les enseignants, les chercheurs, la société en général.

Les apprenants

Nous concentrons d'abord notre effort sur les élèves du supérieur mais la plateforme doit rester capable de s'adapter aux besoins du secondaire ou du primaire.

Quelle est la demande de nos étudiants ?

Des **ressources**

- de qualité en quantité
- alignées avec le cours
- avec des corrections, des évaluations et des commentaires
- pas rébarbatives
- efficaces

Des **activités** didactiques

- à plusieurs
- formatives
- pas rébarbatives
- efficaces
- alignées

Des **indicateurs**

- d'objectifs (la cible, le niveau à atteindre)
- de progression (j'avance ?)
- de niveau (je pourrai atteindre la cible)
- d'efficacité (je fait bien ou mal, trop ou trop peu)

Bien sûr nos étudiants sont des personnes qui ont toujours connu un monde avec une forte présence de l'informatique et d'Internet, leurs exigences sur le look, l'ergonomie, la fluidité et la pertinence sont extrêmement fortes.

Les enseignants

Population très hétérogène quant à leur l'appétence pour l'utilisation du numérique, les enseignants sont toutefois les prescripteurs de solutions. Comme les étudiants, ils ont besoin de ressources, d'activités et d'indicateurs. Mais ils ont un besoin beaucoup plus pressant qui est de construire le cours qu'ils doivent enseigner. Ceci selon deux modalités, soit avec un programme déjà défini par le ministère, soit ex nihilo.

Quelle est la demande de nos enseignants ?

De faire la classe

- définir les **objectifs** du cours, de la semaine, de la séance, du devoir, etc
- **planifier** le travail (communiquer les consignes, définir les échéances, etc)
- **tableaux de bord** (notes, présences, etc)
- **faire cours!** (amphi interactifs, classes, travail de groupes, devoir maison, ..)
- des **outils** : rattrapage, tp live, révision de cours, devoir maison,
- des **évaluations** examen, contrôle continue, validation d'objectifs de formation

De gestion de ressources

- un outil d'édition de cours
- un outil de **création** de ressources
- un outil **d'adaptation** des ressources
- un outil de **recherche** de ressources
- un outil d'éditorialisation ou **curation** des ressources
- un outil de **partage** d'expériences

D'analyse

- analyse des **comportements** pour du conseil
- analyse des résultats -> pilotage des thématiques du cours en fonction de la progression du groupe classe.
- qualité des exercices -> identifier les exercices de chute qui identifient des problématiques.
- qualité des activités -> rébarbatives : gestion du flow

Les chercheurs

Nous parlons ici de chercheurs en science de l'éducation ou des enseignants consciencieux qui souhaitent qualifier leur travail et veulent s'assurer de l'efficacité des modalités pédagogiques qu'ils mettent en œuvre, bien qu'à notre connaissance la difficulté de la

validation des effets de modalités pédagogique soit quasiment insurmontable dans la plupart des cas.

Quelle est la demande de nos chercheurs ?

- pouvoir construire des **expériences**
- pouvoir **choisir et écrire** eux même les indicateurs
- pouvoir **enregistrer** et **vérifier** des expériences

Les institutions

Les institutions ayant depuis longtemps consacré leur énergie et leur moyens à répondre à leur propre besoins : gestion des inscriptions, modélisation des cohortes, remontée des notes, formalisations des programmes, etc. Il ne nous est pas apparu nécessaire de se substituer aux logiciels existants et donc nous n'avons qu'un seul besoin c'est **l'interopérabilité** avec les **logiciels institutionnels** et les **LMS** en général. Notre capacité de développement étant réduite, nous n'aborderons pas ces sujets. Les développements liés à des financements sur projet de recherche ont en général une espérance de vie proche de celle du projet, car la fin du projet implique la fin des financements et donc la fin de la maintenance et donc la mort du logiciel.

Quelles réponses aux besoins exprimés ?

Premièrement y a t'il un logiciel ou une plateforme répondant à nos besoins ?

Pas vraiment¹, soit la créativité est bridée par une ensemble réduit de modèle d'exercices, soit la technologie informatique utilisée est cause de soucis, soit la *programmation* des exercices est complexe, soit l'aspect *libre* n'est pas assuré. La pérennité des logiciels qui sont présentés dans la littérature des sciences de l'éducation est très faible. La reproductibilité des expérimentations en est très fortement compromise, ce qui réduit la qualité scientifique de toutes ces productions. Le biais expérimental d'avoir une plateforme spécifique à une expérimentation donnée affaiblit la fiabilité des résultats.

Un constat accablant: le partage et la curation sont très faiblement mise en avant par les solutions proposées². Souvent les créateurs ne se connaissent pas. Les modalités de publication sont soit trop complexes, soit inexistantes.

Le complexité de la production de ressource est un handicap de beaucoup de plateforme³, nous devons résoudre ce problème.

Quel sont les point clés :

- Promouvoir la réutilisabilité.
- Assimiler toutes les ressources assimilables ou interopérables.
- Prévoir l'assimilation par autrui de nos ressources.

¹ Vous trouverez en fin d'article une liste de sites et d'outils.

² <https://www.francaisfacile.com/> a un système d'identification des modules intéressant avec des tags "top" etc.

³par exemple l'excellente plateforme <https://github.com/UCL-INGI/INGInious> a un modèle d'exercice d'une grande complexité.

- Ressources pédagogiques libres et gratuites.
- Pluridisciplinaire, pouvoir s'adapter aux besoins spécifiques d'une discipline.
- Corrections, validations et créations des ressources effectuées socialement et par les pairs.
- Aucune restriction de publication, tout le monde peut produire.
- Mettre en place un lieu numérique permettant à la collégialité de produire à pleine puissance, profiter de la puissance de l'intelligence collective.
- Proposez autant de niveau de "programmation" que nécessaire dans les ressources.
- Respecter la RGPD, l'éthique et la déontologie des métiers de l'enseignement.

Fort de notre première itération : "PremierLangage" ou nous avons pu tester en vraie grandeur avec 1600 étudiants sur une dizaine d'enseignements la fiabilité de notre premier développement. Nous avons conçu une nouvelle version plus ambitieuse.

Les critères de qualité

Nous souhaitons mettre en œuvre et valider des critères de qualité qui ne sont pas précisés par les besoins des parties prenantes. Ces critères plus génériques sont basés sur une expérience de développement logiciel .

- totalement ouvert et collaboratif : intelligence collective
- pouvant tenir la charge des grosses et très grosses promotion
- pouvant accueillir des expérience de recherche en science de l'enseignement et conserver les données de ces expériences
- Centre de connaissance à la stackoverflow, valorisation par la compétence
- un langage de création à la fois puissant (tout est possible) et accessible (formulaires web pour la plupart des modèles).

C'est simple on veut :

- google pour trouver les documents,
- wikipedia pour la curation
- stackoverflow pour l'entraide
- facebook pour la mise en relation des créateurs
- python pour la programmation
- des thèmes graphiques variés pour l'ergonomie
- des activités interactives

On ne veut pas :

- uniquement travailler par formulaire
- Ecrire dans des langages de programmation pénibles : html, php, css, phtml, OEF etc
- Avoir des syntaxes multiples qu'il faut mélanger entre le visuel, le graphique, le calcul scientifique, les aspects logiques, etc
- avoir à écrire plus de code que nécessaire (réutilisation, extensibilité),
- une interface non modifiable

2. Architecture logicielle

Les problèmes fondamentaux que l'architecture doit résoudre sont la sécurité, la montée en charge et la capacité évolutive.

La sécurité

Utiliser un framework de développement dont la qualité est assurée, nous avons choisi le framework django, d'une part pour ces qualités reconnues de framework sécurisé et pour sa capacité à supporter la montée en charge. De plus, il est nécessaire de nous protéger contre des exécutions du code des étudiants mais aussi du code des enseignants créateurs, en effet pour assurer la capacité évolutive nous devons permettre aux créateurs de ressources d'ajouter du code non sécurisé et non professionnel dans la plateforme. Pour résoudre ce problème de sécurité avec du code potentiellement dangereux (bien entendu à l'insu du créateur ...), nous allons utiliser une approche "banc à sable" c'est à dire avoir une modalité d'exécution du code dans un environnement d'exécution qui nous assure une totale immunité pour le reste du code.

C'est pourquoi nous avons un serveur sandbox qui encapsule la création de machine virtuelle dans lesquelles s'exécute les codes dangereux. La modalité est d'utiliser des conteneurs docker⁴, les processus qui s'exécutent voient leur temps de calcul, leur consommation mémoire et leur consommation disque capés de façon stricte. Le coût en ressource bien qu'important n'handicape pas la montée en charge car il est par construction facile de distribuer les machines exécutant les conteneurs.

La montée en charge

Pourquoi la montée en charge est importante est importante. Nous partons du principe que nous souhaitons un site unique pour permettre l'effet de groupe et l'intelligence collective. D'autre part, nous envisageons des usages sur des promotions de plusieurs milliers d'élèves réalisant les mêmes activités. Les capacités de stockage sont associées aux serveurs sandbox et peuvent être affectées à une formation ou une institution.

La capacité évolutive, maintenabilité, innovation

Pourquoi cette qualité est primordiale ? Car c'est la cause de l'échec de nombreuses solutions logicielles précédentes. C'est un écueil qui a de nombreuses formes:

- des technologies informatiques verrouillées ou spécifiques qui bloquent l'évolution logiciel, la maintenance devient coûteuse l'évolution impossible.

⁴ Les conteneurs docker : <https://docs.docker.com/>

- des formats de données complexes qui rendent l'exportation ou la traduction difficile ou impossible.
- des langages d'expression des ressources très techniques imposant un coût d'entrée important et une maintenance coûteuse.

Premièrement une décomposition en micro-services qui permet de changer les technologies utilisées facilement.

Un format de base qui est un dictionnaire (clés,valeurs) et pour les valeurs des formats **ouverts** et surtout facilement transformable car trivialement lisibles par un programme:

- **markdown** (format universel d'écriture de documentation⁵,
- yaml et json qui permettent de représenter dans un fichier texte lisible des informations que l'on peut projeter en mémoire.
- **python3** et **javascript** pour le code informatique car il est à la fois accessible aux débutants (ici des enseignants) et fournis au programmeurs chevronnés la boîte à outils la plus complète,

Le choix de markdown et python a été imposé par la nécessité de rendre le coût d'entrée raisonnable (nous constatons que les enseignants d'informatique trouve le coût d'entrée acceptable ;). Mais cela reste compliqué pour répondre à une demande d'une approche encore plus simple nous proposons deux innovations, héritage et template.

L'héritage permet de "réutiliser" toutes les clés d'une autre ressource et d'en modifier que quelques-unes pour l'adapter à nos besoins particuliers. Ceci permet une composition des qualités de plusieurs exercices en un seul, cela réduit la quantité de code à écrire, cela structure les types d'exercices et d'activités.

Les template est la possibilité qui est fournie aux programmeurs de transformer (par un code déclaratif) l'héritage d'une ressource en formulaire html. Ainsi après avoir mis au point un exercice proposant des comportements graphiques vous pouvez proposer à d'autres de l'utiliser en ne changeant que l'énoncé et les valeurs d'initialisation. Par exemple un exercice ou l'on doit répondre par le positionnement d'une droite sur un plan orthonormé. Le template vous permettra de poser des questions comme: quelle droite passe par les points A(.,.) et B (.,.) ou dessiner la droite de pente 3 passant par l'origine, etc. Et ceci sans vous préoccuper des modalités d'affichage. Ainsi, progressivement, la plateforme proposera des activités de plus en plus complexes avec des interfaces de programmation de plus en plus simples. Notre objectif est de mettre en place un mécanisme d'amélioration continue et l'innovation ou les usages et les inventions nous surprendront.

La vision de notre rôle est de permettre à nos successeurs de nous dépasser.

Comment faire pour réaliser des expériences pédagogiques avec PLATON

⁵ Le logiciel pandoc, ainsi que des bibliothèques en javascript,java,python,php, permettent de traduire le markdown en tout format d'impression ou d'affichage, en particulier html, latex, word, pdf, etc.

Platon permet d'écrire de nombreux types d'exercice: questions à choix multiples, questions ouvertes, classement et ordonnancement, programmation, réflexes, mémorisation. Il est possible d'importer de nombreux formats d'exercices quizz, GIFT, CLOZE, exo7, certains exercices WIMS. Il est possible d'écrire ses propres tableaux de bord personnalisés (pour le pilotage de la classe). Platon propose un système d'exportation de logs (il est possible pour les chercheurs d'avoir les logs bruts RGPD) et utilise des technologies LTE qui permettent de produire les données pour traitement comme par exemple un cube olap.

La capacité d'adaptation de la plateforme en fait un outil permettant la réutilisation. La programmabilité permet d'adapter par exemple: les visuels à des enfants, la dynamique à de l'apprentissage par expérimentation/simulation. D'inventer de nouveaux algorithmes, de mémorisation, des ateliers d'évaluation par les pairs, de la dynamique de groupes etc. De travailler sur les aspects métacognitifs, l'engagement, etc. De travailler à des projets de neurosciences pédagogiques ou didactiques.

L'intérêt de développer les expériences sur la plateforme Platon est l'aspect ouvert des productions qui permet de partager les outils tout en autorisant des expériences privées. Les données non anonymisées sur les apprenants étant privées et réservées aux enseignants directement associés aux apprenants et par dérogation aux chercheurs.

Enfin les bonnes pratiques une fois mise en œuvre de façon expérimentale pourront être entendues à tous les utilisateurs de la plateforme. Le système de communication et de curation des ressources ajoutera à l'analyse scientifique des questions comme la facilité de prise en main par les formateurs et les apprenants et les impacts indirects comme le fait qu'une expérience dans une discipline a parfois un impact négatif sur l'apprentissage dans une autre (les étudiants étant absorbés par l'investissement nécessaire d'un cours inversé dans la discipline principal de la formation, expérience vécu).

Une expérience

La première itération de PLaTon se nomme PremierLangage

Depuis quatre ans, nous déployons des exercices dans la licence math-info de l'université Gustave Eiffel. Fort d'un renouvellement de déploiement en seconde année de licence pour la matière architecture des systèmes informatiques, nous avons proposé 113 exercices d'informatique système aux 120 élèves de la promotion ayant choisi la branche informatique du cursus.

PremierLangage s'avère dès sa seconde année d'utilisation un outil incontournable. Premièrement, c'est le premier cours d'informatique qui exige aux élèves d'avoir accès à une console Unix (une difficulté pratique en dehors de l'université), or PremierLangage permet l'écriture d'exercice qui simule un terminal Linux. Et ces exercices interfaçant un terminal sont web-responsive. Ainsi, avant même que les élèves arrivent à installer le système d'exploitation pédagogique Linux, ils peuvent déjà travailler depuis leurs téléphones, tablettes ou ordinateurs sous leurs systèmes et navigateurs web préférés. Le distanciel a très fortement dégradé ce cours, avant, les élèves ramenaient leur ordinateurs portables en TP pour avoir de l'aide à l'installation de Linux. Mais PremierLangage a annulé

cette dégradation car il permet une formation des jeunes au nouveau système d'exploitation avant qu'ils arrivent à l'installer sur leur matériel informatique personnel en autonomie.

Les 113 exercices ayant été partitionnés en cinq domaines de notions, tout comme les 7 exercices de l'examen final. Il a pu être calculé l'influence de la plateforme sur la réussite des apprenants (une corrélation claire: le fait d'utiliser la plateforme suffisamment est un gage de réussite). PremierLangage est un support supplémentaire non obligatoire mais il s'avère, chiffres à l'appui, qu'il constitue la meilleure stratégie de révision. En effet, la plateforme permet l'extraction en direct des notes dans ces formats particulièrement interopérables (comme csv, directement exploitable dans tous les tableurs raisonnables). En une ou deux heures, l'enseignant a pu estimer avec les traces de ces élèves la couverture des exercices déployés par rapport au contenu du programme, l'influence du travail sur la plateforme et la réussite à l'examen final et enfin l'alignement pédagogique entre les ressources déployées et l'examen sommatif final de la matière.

Une conclusion

L'objectif de la plateforme est de proposer un outil pour les apprenants et les enseignants. Notre philosophie est qu'un bon enseignant avec donc de bons outils aide très efficacement les apprenants. Un bon outil est un outil qui réduit le coût des opérations répétitives peu productives. Donc nous tâchons de permettre une programmation facile, un suivi automatisé avec des alertes, une capacité accrue pour l'apprenant de bien cibler les objectifs et la démarche pour les atteindre en un minimum de temps et d'efforts (les efforts seront toujours nécessaires ;).

Le projet est pour le moment dans son enfance bien que des étapes de développement soient réalisées il reste des grandes zones à concevoir et à implémenter. La multiplication des niveaux de programmation et de conception de ressources, les besoins en curation et éditorialisation font que toutes les contributions seront les bienvenues.

Le développement de PLaTon est financé depuis 2013 par l'université Gustave Eiffel et un projet IDEA⁶. Depuis 2 ans par un projet AMI WIMS-EVO⁷. Le projet WIMS-Ed a contribué en rémunérant un apprenti pendant deux ans.

Remerciements

Tout d'abord remercions les étudiants et apprentis ayant contribué au développement de Premier Langage, Romain Girard, Christophe Callé, Jérémie le Bastard, Fanny Garnevault, Stéphane Sanchez, Anna Diaw, Killian Blanchard, Yohan Rameau, Thomas Juillard. Particulièrement les remerciements vont à Quentin Coumes qui a construit le premier environnement de développement de Premier Langage et professionnalisé les modalités de participation au développement.

⁶ <http://idea.univ-paris-est.fr/fr/presentation-idea/document-1556.html>

⁷ <https://wimsedu.info/?p=5424>

Particulièrement les remerciements vont à Mamadou Cissé, qui a apporté sa maîtrise du framework Angular, ce qui a transformé complètement la stratégie générale de conception de PLaTon.

Des enseignants en premier l'équipe rapprochée Magdalena Kobylański, Nicolas Borie, David Doyen, sans qui le projet n'aurait pas existé.

A Damir Buskulic et Etienne sandier pour leurs idées et leur enthousiasme, aux équipes de L1/L2, Olivier B, Olivier C, Marie, Justine, Wenjie, Marie-Pierre, Marc, Philippe, Nadim, Hugo, Frederic, Emmanuel,

```
@manual{PLaTon-web,  
  author = {{R}evuz, {D}ominique and others},  
  title = {PLaTon, a Platform for Learning and Teaching ONline},  
  year = 2020,  
  note = {hosted by github at https://premierlangage.github.io/PLaTon-web/}}  
}
```

```
@manual{PLaTon-doc,  
  author = {{R}evuz, {D}ominique and others},  
  title = {Documentation de PLaTon côté enseignant},  
  year = 2020,  
  note = {hosted by readthedocs at https://documentationpl.readthedocs.io/fr/latest/}}  
}
```

Quelques sites en ligne ou d'outils que l'on peut mettre en ligne.

Français facile <https://www.francaisfacile.com/> <https://micetf.fr/>

WIMS <https://wims.u-pem.fr/wims.cgi?module=help%2Fmain>

extension de moodle: VPL <https://vpl.dis.ulpgc.es/>

Feuille d'exercices: <https://www.onlinemathlearning.com/math-worksheets.html>

<https://exo7.emath.fr/>

liste d'outils <https://www.ispringsolutions.com/blog/free-teaching-tools>

jeux mathématiques: <https://www.mathplayground.com/math-games.html>

<https://www.destinenseignante.ca/2020/04/outils-conseils-enseignement-distance.html>

Les outils Google ou EDX.

Des sites sur github : exercism

outil de gestion de devoir d'informatique: https://gitlab.info.uqam.ca/info/travo_git

<https://learningapps.org/>

<https://quizlet.com/fr-fr>

<https://fr.educaplay.com/types-d-activites/>