

# La ludification pour l'apprentissage de la programmation : exemple avec le langage Python

JACOBS Alexandre<sup>1</sup>

Superviseur : MASSART Thierry<sup>1</sup>

<sup>1</sup>Université Libre de Bruxelles

alejacob@ulb.ac.be

tmassart@ulb.ac.be

**Résumé**—Gamification, ou Ludification en français, est une expression basée sur le mot « game », elle consiste en l'utilisation du jeu pour transmettre un message ou améliorer une offre. Dans notre cas, nous étudierons l'apprentissage d'un langage de programmation qui se situe bien dans le contexte ludique puisqu'un jeu vidéo est conçu avec des langages de programmation. Dans cet article, nous traiterons d'une part de l'état de l'art avec les études réalisées sur le sujet et des sites mettant en œuvre la ludification pour l'apprentissage d'un langage de programmation. D'autre part, nous verrons quels types d'activités ludiques intéresseraient les étudiants et nous discuterons de la meilleure utilisation possible d'un site permettant l'apprentissage d'un langage de programmation de manière ludique.

seignement peut amener aussi des difficultés pour certains étudiants.

Dès lors, les solutions à cette problématique pourraient être un apport d'aide (qui est déjà présent) et un remaniement de l'enseignement avec l'introduction de la "gamification".

Dans la suite de l'article, nous verrons l'état de l'art, en quoi l'enseignement avec la gamification peut être utile, quels moyens sont mis en œuvre, ce que les étudiants en pensent et comment employer au mieux un site utilisant la ludification pour un apprentissage d'un langage de programmation tel que Python.

## I. INTRODUCTION

Le but de cet article est de résumer l'état de l'art existant et d'évoquer les possibles problématiques et bienfaits de "la ludification pour l'apprentissage de la programmation".

On définit la ludification comme le fait d'appliquer des mécanismes, concepts à ce qui ne relève pas du jeu.

De nos jours, la technologie évoluant vite et induit un engouement pour les études se rapportant aux sciences informatiques et à la programmation. On y retrouve des étudiants venant de différents horizons, qui ont des bagages intellectuels (formations, compétences académiques) fort variés. Cela conduit à un problème d'inégalité entre les élèves, par conséquent certains décrocheront plus vite que d'autres. De plus, la méthode traditionnelle d'en-

## II. ÉTAT DE L'ART

Concernant les articles, qui sont en généraux des cas d'étude sur l'utilisation de la ludification, il en ressort que cela a un impact bénéfique pour les étudiants. En effet, ceux-ci semblent plus investis et plus curieux dans l'apprentissage d'un langage de programmation, l'amélioration de concepts et de compétences.

Dans l'un des articles, Rais et al. (2011), qui fait état d'une approche en milieu éducatif, la manière de faire comprendre au mieux un concept informatique a été celle utilisant les logiciels « Alice 2.0 », « GAPS 1.0 », « Greenfoot » et « Darwin » à travers des réalisations d'étudiants en petits groupes afin de mettre en situation l'orienté objet. Par ailleurs,

l'auteur dit que le fait d'intégrer les jeux dans un tutoriel en ligne pourrait mener à une bien meilleure compréhension. Pour finir, il en ressort également que les difficultés que l'étudiant rencontre qui sont souvent dues au manque de « guidance » (aide) lors des sessions d'exercices.

Dans une autre étude ([da Silva and da Silva Aranha (2015)]), on fait part d'un même engouement où jusqu'à 75% des étudiants qui y ont pris part, ont remis toutes les activités. De plus, il nous indique aussi que 100% des étudiants qui ont participé à l'étude, se disent prêts à y participer à nouveau. Ici, la manière d'enseigner était des cours magistraux, des podcasts et une utilisation de «Construct2».

De tous les articles lus, on peut tirer la conclusion que la ludification amène de réelles choses bénéfiques, du fait par exemple de permettre de conceptualiser d'une meilleure manière les concepts et langages de programmation. Pour les articles ayant des résultats, on peut remarquer qu'il y a une nette augmentation de la compréhension de certains concepts par rapport à une méthode traditionnelle d'enseignement. De plus, on peut aussi voir au travers de l'utilisation de la gamification que la communication se fait plus aisément entre étudiants. Par ailleurs, l'utilisation de la ludification permet aussi aux étudiants de développer un esprit critique et une pensée informatique.

Tous les sites web qui sont décrits ci-dessous sont tous des plateformes alliant le jeu à l'éducation en mode solo ou en multijoueurs. Ils permettent presque tout l'apprentissage des rudiments d'un langage tel que Python (Code Combat) et ils permettent aussi d'améliorer ses compétences.

#### A. Code Combat

Code Combat<sup>1</sup> permet, en plus d'apprendre les bases ou de s'améliorer en Python, de développer d'autres compétences dans d'autres langages de programmation.

1. <https://codecombat.com/>

Concernant le design du site, celui-ci dispose de plusieurs mondes. Un monde est composé de plusieurs niveaux qui représentent chaque aspect du concept développé dans le monde choisi, tel que les boucles, fonctions, variables, etc. De plus, chaque niveau a sa propre difficulté(cf. Appendices : Figure 7).

La plateforme d'un niveau est agencée de la façon suivante (cf. Appendices : Figure 8), elle contient 2 parties, celle de gauche qui est une représentation graphique des actions que la personne codera dans la partie de droite. Elle contient dans une petite zone l'objectif que la personne doit atteindre pour pouvoir accéder aux différents niveaux suivants. La partie de droite est une zone d'éditeur de texte où l'on met le code qui permet la réalisation de l'objectif du niveau. Dès que la personne décide d'exécuter son code, la partie graphique de gauche se met en marche pour afficher un rendu du code.

Les premiers niveaux sont conçus d'une manière assez simple pour que les personnes qui n'ont jamais codé auparavant puissent comprendre facilement les concepts. De plus, le site permet à des professeurs de pouvoir créer des classes afin d'y regrouper leurs élèves. Cela permet de voir la progression de ces derniers et de repérer les possibles incompréhensions afin d'y remédier lors des cours théoriques ou séances d'exercices.

Par contre, un désavantage majeur est le fait que si un professeur crée une classe, celui-ci devra déboursier de l'argent pour acheter des licences afin de permettre d'accéder à d'autres mondes que celui d'introduction. Idem pour les personnes qui sont inscrites comme étudiant et ne faisant pas partie de classe.

#### B. Checkio

Checkio<sup>2</sup> est destiné à la ludification et contient également la possibilité de créer des classes pour des professeurs le désirant. Tout comme pour Code Combat, il faudra passer à la caisse pour pouvoir

2. <https://checkio.org/>

utiliser cette fonctionnalité. Par contre, il est possible de rédiger ses propres problématiques pour ses élèves.

Le site contient divers mondes (planètes), chaque monde est constitué de plusieurs niveaux de difficultés différents qui abordent divers aspects du langage de programmation Python. Les niveaux ont chacun un thème comme les problèmes de mathématiques ou d'autres types de problèmes qui peuvent être résolus par un programme.

En accord avec ce qui est dit dans l'article de Hoarau (2017), Checkio.org ne met pas en avant le côté graphique comme sur Code Combat ou Codingame et il n'y a pas d'animations lors de l'exécution du code.

### C. Codingame

Codingame<sup>3</sup> est une plateforme qui permet de coder en Python et dans plein d'autres langages. Au début, ce site n'était pas conçu pour permettre l'apprentissage par ludification, mais devait juste servir à tester des candidats pour un poste dans une entreprise de développement informatique. Le recrutement était possible si on arrivait à résoudre des problèmes, cela permettait ainsi de trouver le job qui convient le mieux et la personne la plus qualifiée en même temps [Hoarau (2017), Ivanova (2016)].

La plateforme propose différents problèmes sous forme de puzzles ou bien de challenges avec des modes solo et multijoueurs. Ceux-ci vont du niveau le plus simple au plus compliqué.

La présentation d'un puzzle (cf. Appendices : Figure 9) est du même style que la plateforme Code Combat et comporte deux parties. Cependant, dans la partie de gauche, juste en-dessous de la représentation graphique de l'exécution du code fourni, se trouve l'objectif du puzzle avec des règles, des indices quant à la façon de le résoudre.

3. <https://www.codingame.com/>

L'autre partie de droite est presque identique à celle de Code Combat, mais diffère par le fait qu'en dessous de la zone d'édition du code, se trouve le nombre de tests que doit passer le code avec la possibilité d'afficher ce que contiennent ces tests avec les valeurs d'entrées et les différents résultats attendus. Quand le code aura été exécuté, la zone montrant le nombre de tests changera pour montrer ceux qui ont été réussis ou non. La présentation d'une page d'un Challenge est identique à celle des puzzles, la seule différence est que nous sommes en compétition avec d'autres personnes.

Codingame présente un design attirant avec de beaux graphismes de jeux, et des problèmes assez diversifiés. Cependant, ce dernier vise plutôt un public ayant déjà des bases en programmation, il est donc à utiliser après en avoir appris les rudiments avec un autre site utilisant la ludification.

Par ailleurs, l'équipe de développeurs du site sont au courant de ce « problème », ils planchent sur une solution qui permettra cet apprentissage qui est d'ores et déjà disponible sur un site, Tech.io, mais en version BETA.

De plus, Codingame propose un tutoriel<sup>4</sup> détaillé à l'attention des professeurs voulant introduire les principes de base de la programmation par la ludification. Le site permet aux personnes qui le désirent de créer leur propre puzzle afin de transmettre leur savoir et de permettre à d'autres personnes de s'améliorer avec ceux-ci.

### D. Code Hunt

Code Hunt<sup>5</sup> est un site mettant en œuvre la ludification (gamification), il s'agit donc d'un site à la portée de tous qu'on soit programmeur expert ou novice. Il n'est cependant disponible actuellement que pour C# ou JAVA et non pour Python qui est pourtant un des langages conseillés lorsqu'on commence à apprendre la programmation. Nous

4. [http://files.codingame.com/codingame/learning/CodinGame\\_for\\_Teachers\\_Onboarding\\_Lesson\\_2.pdf](http://files.codingame.com/codingame/learning/CodinGame_for_Teachers_Onboarding_Lesson_2.pdf), <https://www.codingame.com/blog/1-hour-to-teach-programming-with-codingame-in-fun-way/>

5. <https://www.codehunt.com/>

avons quand même choisi d'en parler, car celui-ci utilise lui aussi de manière agile la gamification. C'est un site qui dispose de plusieurs niveaux et de mondes (cf. Appendices : Figure 10). On peut aussi remarquer que les différents mondes abordent les différents concepts de base de la programmation générale.

Un niveau est constitué de deux parties (cf. Appendices : Figure 11) : un espace pour développer son code dans une sorte d'éditeur de code et de l'autre côté, un tableau contenant les résultats attendus et le résultat que vous produisez en sortie avec le code de la partie de gauche. Ce tableau permet donc de voir si notre code est correct ou non et il inclut également une barre de progression afin de voir où l'on se situe.

Le but de Code Hunt, en plus de l'apprentissage, est d'amener la personne à développer un code sur la base des entrées des tests qui sont disponibles dans la partie de gauche du site.

Enfin, en montrant différentes entrées testées sur le code proposé, il amène la personne après plusieurs changements à se rapprocher du « code secret » (Meilleure solution du niveau ; cf. Figure 11).

Code Hunt fût créé en partenariat avec Microsoft Research Outreach chez Microsoft Research et le groupe de recherche en génie logiciel (RISE) sur une base Microsoft's cloud Azure. Il est fort utilisé dans le monde, comme l'indique une compétition organisée en Chine avec plus de 13.000 étudiants engagés dans celle-ci [Tillmann et al. (2014)].

De manière générale, les différents concepts en programmations qu'on peut apprendre, voir cette liste ci-dessous, et voici un tableau comparatif avec les différentes plateformes décrites précédemment avec les différents concepts de programmation.

Liste des différents concepts :

- Variables ; qu'est-ce que cela est et comment l'utiliser.
- Conditions
- Instruction
- Types de variables possibles : chaînes de caractères, entiers("integer"), nombre à frac-

tionnaires("float"), listes, tuple, dictionnaires, ensembles, ... Comment définir ces types, les méthodes possibles sur celles-ci et comment les utiliser au mieux dans un programme.

- Boucles : qu'est-ce qu'une boucle while (tant que), for (jusqu'à), comment les utiliser et la différence entre ces deux types de boucles.
- Fonctions : qu'est-ce qu'une fonction, comment les définir et comment les utiliser dans un code.
- Classes : qu'est-ce qu'une classe et son but dans un code, comment les définir et comment les utiliser dans un code.

Concepts \ Plateformes	Code combat	Codingame	Checkio	Code Hunt
Variable	X	X (amélioration)	X	X
Type de variables (entier, float, liste, tuple, dictionnaire, ensemble, ...)	X	X (amélioration)	X	X
Conditions	X	X (amélioration)	X	X
Boucles	X	X (amélioration)	X	X
Fonction	X	X (amélioration)	X	
Classe	X	X (amélioration)		

FIGURE 1. Tableau comparatif des plateformes en fonctions des concepts qui sont possible d'apprendre.

En résumer avec ce tableau comparatif, nous pouvons remarquer que majoritairement presque toutes les plateformes ayant vues précédemment, elles permettent un apprentissage de toutes voir presque toutes des différents concepts détaillés ci-dessus. Concernant Codingame, on peut voir la mention "amélioration" est présente du fait que cette plateforme est plutôt pour de l'amélioration de concepts que pour de l'apprentissage.

Pour conclure cet état de l'art, ces différents sites détaillés ci-dessus ont tous comme objectif de promouvoir l'apprentissage de la programmation et l'amélioration de ses compétences par la ludification. De plus, ceux-ci ne sont pas seulement pour un enseignement personnel, mais aussi pour un enseignement à plus large échelle. Ces plateformes ont un réel impact, car elles ont chacune plus ou moins des dizaines de milliers d'utilisateurs, voire même beaucoup plus pour Codingame qui est une plateforme assez connue. Elles encouragent donc les personnes, les motivent à coder et à apprendre la manière d'agencer au mieux leur code. Elles permettent aussi de se forger un esprit critique et informatique quant à la manière de résoudre un problème.

Par ailleurs, il existe d'autres alternatives utilisant la ludification grâce à la remise de points et de badges lors de la réussite d'une leçon comme Codecademy. Ces sites sont interactifs avec des explications théoriques, des mises en situation et des quiz pour certains. Ils permettent tous d'apprendre Python.

### III. PRÉSENTATION VIDÉO

Concernant l'utilisation d'un site web de gamification, nous avons réalisé une vidéo sur les sites Code Combat et Codingame.

Les différents sites présentés dans cette vidéo permettent, comme dit dans cette dernière, l'amélioration de concepts (Codingame) et l'acquisition des concepts de base de programmation en Python aux concepts plus complexes (Code Combats). Code Combat permet vraiment à l'aide des différents problèmes et niveaux de passer sur ces différents concepts :

- Variables
- Conditions
- Instruction
- Types de variables possibles : chaînes de caractères, entiers("integer"), nombre à fractionnaires("float"), listes, dictionnaires, ensembles, ...
- Boucles
- Fonctions
- Classes

Le choix s'est porté naturellement d'une part sur Code Combat qui permet réellement, comme vu dans cette présentation vidéo un apprentissage des différents concepts de base de programmation Python jusqu'aux concepts plus complexes et d'autre part sur Codingame, car il permet une amélioration des concepts de base et complexes via une multitude de challenges, puzzles qui sont du niveau les plus simples au plus compliqués.

La présentation de cette vidéo étant disponible à ce lien.<sup>6</sup>

6. <https://youtu.be/WGoEsYjFTtE>

### IV. SONDAGE

Nous avons réalisé un sondage afin de savoir comment était connu la ludification parmi des étudiants de 1re année de bachelier en sciences informatiques et de l'École Polytechnique de Bruxelles, de savoir les principales causes d'échecs et de difficulté et aussi leurs envies et attentes concernant la ludification quant à son utilisation dans le cours d'introduction à la programmation « INFO-F101 » dispensé par le Pr. Thierry Massart.

Les questions de l'enquête furent :

- Aviez-vous déjà entendu parler de la ludification comme aide pour l'apprentissage ?
- Aviez-vous déjà essayé une plateforme web mettant cela pratique pour l'apprentissage de la programmation ?
- Avez-vous abandonné le cours de programmation INFOF101 ou INFOH101 ?
- Quelles en sont les causes, facteurs de cet abandon ?
- Est-ce que la ludification mieux utilisée, aurait pu permettre de ne pas abandonner le cours ?
- Bien que le concept de ludification soit déjà présent dans le cours d'initiation à la programmation, par exemple dans les projets à réaliser, est ce qu'une intégration de manière encore plus présente, par exemple lors des TP's via des jeux ou utilisation de plateforme web, vous permettrait de relier la théorie aux exemples concrets ?
- Quels sont les types de difficulté que vous avez rencontré lors de ton apprentissage de python ?
- Est-ce que ces difficultés auraient pu être améliorées par l'utilisation de la ludification avec un site tel que Code Combat ou bien avec un autre support utilisant la ludification ?
- Qu'est-ce que vous attendez de l'utilisation de la ludification ?
- Pensez-vous que le cours/exercices serait plus attirant avec la ludification plus présente que maintenant ?

Dans ce sondage, une centaine d'étudiants y ont pris part. Concernant les résultats à ces différentes questions, on remarque aisément que la plupart des étudiants ont déjà entendu parler de cela, mais à contrario jamais utilisé un site web tel que Codingame, CodeCombat et autres (cf. Figures 2 et Figure 3).

Aviez-vous déjà entendu parler de la ludification comme aide pour l'apprentissage ?

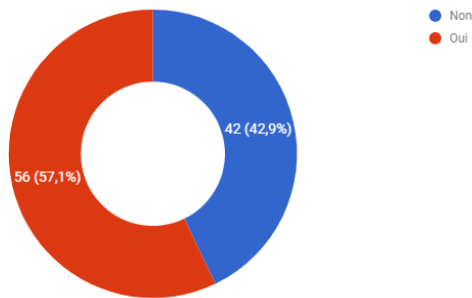


FIGURE 2. Pourcentage d'étudiants ayant déjà entendu parler de la "Ludification".

Aviez-vous déjà essayé une plateforme web mettant cela pratique pour l'apprentissage de la programmation ? (Parmi ceux ayant déjà entendu parler de la ludification)

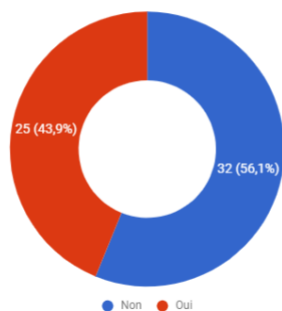


FIGURE 3. Pourcentage d'étudiants ayant déjà essayer une plateforme avec de la "Ludification".

Aussi, on remarque que le taux de découragement (abandon, cf. Figure 4) parmi les sondés est faible, et que les causes de celles-ci sont souvent des cours fort théoriques et un découragement des étudiants face à la quantité de travail à connaître et à fournir.

De plus, les difficultés récurrentes que les étudiants rencontrent sont que les exercices sont abstraits, incompréhension de certains points de matières (boucles, fonctions, classes, ...) pour certains et aussi dû à un nombre grandissant d'étudiant en 1re et moins d'aide de la part des assistants lors de

Aviez-vous abandonné le cours de programmation INFOF101 ?

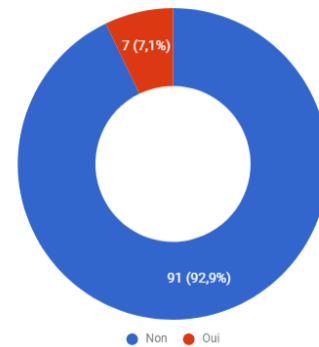


FIGURE 4. Pourcentage d'étudiants qui ont abandonné le cours d'introduction à la programmation.

questions spécifiques par des étudiants de manière individuelle.

Est-ce que ces difficultés auraient pu être améliorées par l'utilisation de la ludification avec un site tel que Code Combat ou bien avec un autre support utilisant la ludification ?

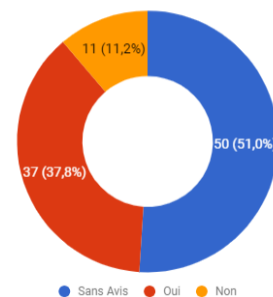


FIGURE 5. Pourcentage d'étudiants qui trouvent que la ludification permettrait de résoudre des difficultés d'apprentissage.

De plus en les interrogeant sur le fait de savoir si la ludification aurait pu aider à résoudre les difficultés qu'ils rencontraient, cela est un « oui » pour la majorité des sondés(cf. Figure 5).

Par ailleurs, ils pensent aussi que la gamification dans l'apport du cours d'introduction à la programmation et autres est que cela peut donner des exemples plus concrets et intriguer, encourager les étudiants à coder de manière ludique, car la plupart des jeunes étudiants jouent de plus en plus aux jeux vidéo.

En conclusion de ce sondage, on peut déduire que les étudiants sont partant pour une intégration à plus grande envergure de la ludification de ce qui est déjà présente. Les résultats de ce sondage sont

Penses-tu que le cours/exercices serait plus attirant avec la ludification plus présente que maintenant ?

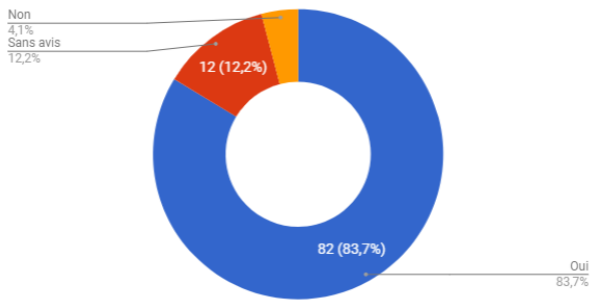


FIGURE 6. Pourcentage d'appréciation parmi les étudiants quant à une utilisation plus intense de la ludification.

disponibles à cette adresse<sup>7</sup>.

## V. CONCLUSION

Pour conclure, cet article apporte un ensemble d'avantages et d'inconvénients concernant l'intégration de la ludification dans l'apprentissage de la programmation. La ludification est un outil qui bien utilisé dans l'apprentissage peut donner goût aux étudiants de découvrir par eux même, et aussi permettre aux étudiants de conceptualiser plus facilement le fonctionnement de programme et l'utilisation des différents concepts de programmations. Comme possibles travaux futurs sur l'utilisation de la ludification dans l'enseignement dans le cadre du bachelier en sciences informatiques peut être une refonte des projets du cours de programmation, l'instauration d'une plateforme telle qu'Upylab, mais plus axé sur la ludification et aussi remanier les projets pour y inclure d'avantage de ludification.

## VI. REMERCIEMENTS

Je remercie tout d'abord Mr. Massart pour ce sujet fort intéressant et utile pour un apprentissage ludique. Par la suite mes parents qui ont été de bon conseil et je remercie également grandement mes amis pour leurs avis externes, relectures et de bons

conseils dans les bons moments ainsi que dans les mauvais.

De plus, je remercie tous les étudiants de 1re en sciences informatique et à l'École Polytechnique de Bruxelles qui ont pris part à ce sondage.

## RÉFÉRENCES

- Azmi, S., Iahad, N. A., and Ahmad, N. (2015). Gamification in online collaborative learning for programming courses : A literature review. *ARP Journal of Engineering and Applied Sciences*, 10(23) :1–3.
- da Silva, T. R. and da Silva Aranha, E. H. (2015). Online game-based programming learning for high school students—a case study. In *Frontiers in Education Conference (FIE), 2015 IEEE*, pages 1–8. IEEE.
- Hoarau, S. (2017). Ludification pour la motivation en apprentissage de la programmation. *Computer Science and Mathematics Laboratory, Université de La Réunion, Saint-Denis, France*.
- Ibáñez, M.-B., Di-Serio, A., and Delgado-Kloos, C. (2014). Gamification for engaging computer science students in learning activities : A case study. *IEEE Transactions on learning technologies*, 7(3) :291–301.
- Ivanova, S. (2016). Learning computer programming through games development. In *The International Scientific Conference eLearning and Software for Education*, volume 1, page 492. "Carol I" National Defence University.
- Kazimoglu, C., Kiernan, M., Bacon, L., and MacKinnon, L. (2012). A serious game for developing computational thinking and learning introductory computer programming. *Procedia-Social and Behavioral Sciences*, 47 :1991–1999.
- Khaleel, F. L., Ashaari, N. S., Meriam, T. S., Wook, T., and Ismail, A. (2015). The study of gamification application architecture for programming language course. In *Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication*, page 17. ACM.
- Knutas, A., Ikonen, J., Nikula, U., and Porras, J. (2014). Increasing collaborative communications in a programming course with gamification : a

<sup>7</sup>. [https://docs.google.com/spreadsheets/d/1cR7VC\\_SFk4XyGPf9ForTGZ6sVbxYnXBBUnQXW0gBB5M/edit?usp=sharing](https://docs.google.com/spreadsheets/d/1cR7VC_SFk4XyGPf9ForTGZ6sVbxYnXBBUnQXW0gBB5M/edit?usp=sharing)

- case study. In *Proceedings of the 15th International Conference on Computer Systems and Technologies*, pages 370–377. ACM.
- Malliarakis, C., Satratzemi, M., and Xinogalos, S. (2014). Educational games for teaching computer programming. In *Research on e-Learning and ICT in Education*, pages 87–98. Springer.
- Mitamura, T., Suzuki, Y., and Oohori, T. (2012). Serious games for learning programming languages. In *Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on*, pages 1812–1817. IEEE.
- Rais, A. E., Sulaiman, S., and Syed-Mohamad, S. M. (2011). Game-based approach and its feasibility to support the learning of object-oriented concepts and programming. In *Software Engineering (MySEC), 2011 5th Malaysian Conference in*, pages 307–312. IEEE.
- Swacha, J. and Baszuro, P. (2013). Gamification-based e-learning platform for computer programming education. In *X World Conference on Computers in Education*, pages 122–130.
- Tillmann, N., De Halleux, J., Xie, T., and Bishop, J. (2014). Code hunt : Gamifying teaching and learning of computer science at scale. In *Proceedings of the first ACM conference on Learning@scale conference*, pages 221–222. ACM.

## APPENDICES

### FIGURES





FIGURE 7. Les différents mondes existant contenant chacun plusieurs niveaux de difficultés différentes.

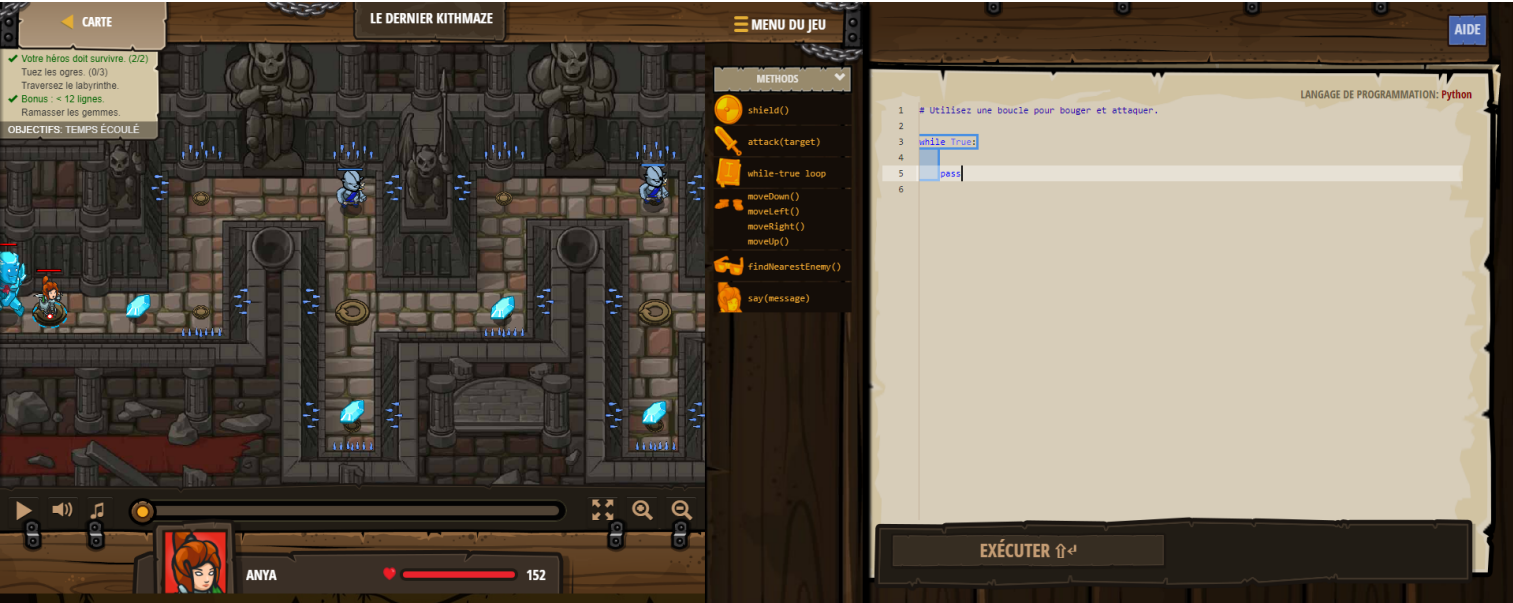


FIGURE 8. Design d'un niveau composé de deux parties.

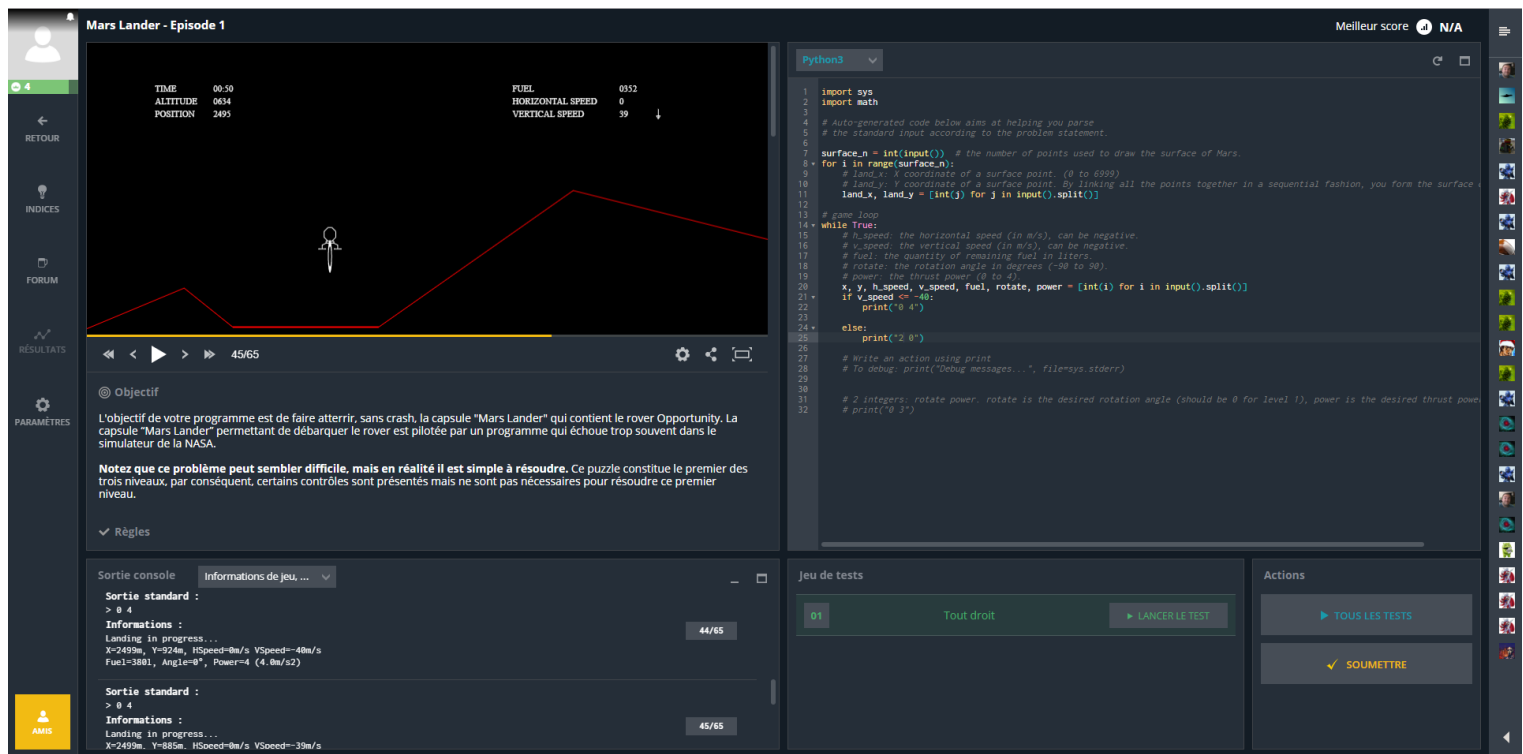


FIGURE 9. Design d'un niveau composé de deux parties.



FIGURE 10. Les différents mondes existant contiennent chacun plusieurs niveaux de difficultés différentes.

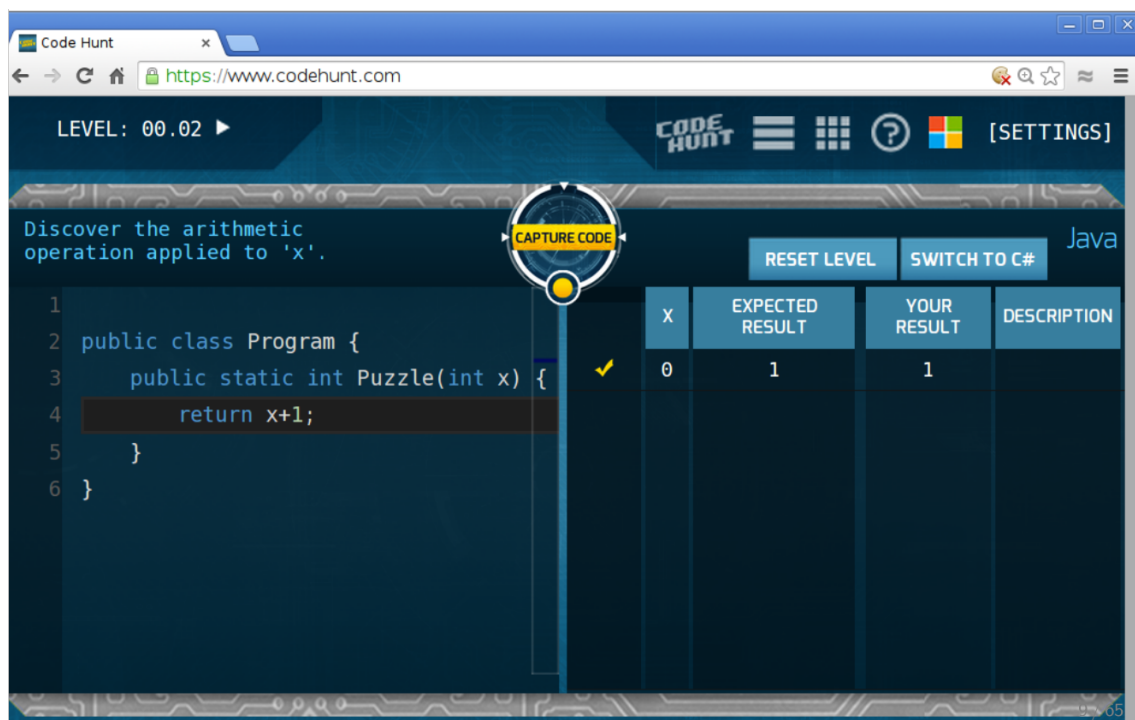


FIGURE 11. Code secret, meilleure solution passant tous les tests.