



Ce document est l'un des livrables à fournir lors du dépôt de votre projet : 4 pages maximum (hors documentation).

Pour accéder à la liste complète des éléments à fournir, consultez la page [Préparer votre participation](#).

Vous avez des questions sur le concours ? Vous souhaitez des informations complémentaires pour déposer un projet ? Contactez-nous à info@trophees-nsi.fr.

NOM DU PROJET : Shroom TD

> PRÉSENTATION GÉNÉRALE :

Chaque année en NSI, nous avons pour consigne d'élaborer un **projet** de notre choix. En revanche, nous avons, cette année, la possibilité de participer aux **Trophées NSI**, ce qui représentait pour nous l'occasion de nous dépasser ~~(et de nous mesurer à la France entière)~~.

Le professeur nous avait proposé dans un premier temps l'idée du **Tetris revisité** par SEGA : Dr. Robotnik's Mean Bean Machine... Idée pour laquelle nous n'étions pas sûr du rendu que nous aurions pu proposer. Nous avons donc décidé de trouver une autre idée qui nous plaisait et nous inspirait bien plus : un jeu de **Tower Defense** sur le thème des **champignons et des slimes** ~~(On remercie Scotty et Jérôme pour leurs idées farfelues)~~. Les champignons faisant partie d'un univers déjà pensé dans un projet précédent (*RPG dans lequel le guide du jeu est un champignon nommé Pablo venant du vieux port de Mare-Sel*).

Le principe est simple, **poser des tours** (champignons) et **éliminer les ennemis** (slimes) avant qu'ils ne franchissent la fin du chemin.

L'intérêt du projet était de créer un jeu aussi original que **complexe** à sa **réalisation**. En effet, **la quasi-totalité du projet a été réalisée par nos propres moyens** (Même les éléments graphiques et les bruitages (*les sessions de vocal étaient bien divertissantes*)). Une très petite partie du projet ne vient donc pas de nous comme les musiques ([Waterflame](#)).

> ORGANISATION DU TRAVAIL :

L'équipe est composée de :

Scotty Pruvost (Le mec jamais satisfait de ses propres créations)

Jérôme Vandewalle (Le mec qui fait toujours une copie locale et fais ses magouilles avant de partager avec le reste du groupe et d'actualiser le répertoire OneDrive)

Mathis Van Uytvanck (le mec qui fait crash le jeu H24 avec ses modifications et ajouts)

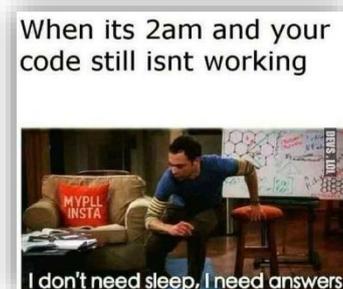
Le travail fut réparti selon la description suivante :

- **Scotty Pruvost** : Directeur Artistique en charge de la création des éléments graphiques (**TOUS** les éléments graphiques), développeur de tous ce qui possède un rapport avec la map et le déplacement ainsi que tout ce qui a un rapport avec les tours (sous-classes des tours, classe map...). Responsable des données JSON (maps, upgrades, waves).
- **Jérôme Vandewalle** : Développeur en charge de l'interaction entre le jeu et le joueur et toutes fonctionnalités liées au gameplay (sons, animations, menu...). Puis la compatibilité du jeu sur les différentes tailles d'écran et sur différents systèmes d'exploitation.
- **Mathis Van Uytvanck** : Développeur en charge des ennemis (slime, boss), leurs points de vie, caractéristiques, résistances. Sur les boss, développement du système de bouclier. Création du système de simplification de vagues pour la lisibilité et l'allègement en structure du fichier « waves.json ».

Aussitôt le projet commencé, Aussitôt il fallait s'organiser, c'est dans cette directive qu'un répertoire **OneDrive** partagé avec les membres fut créé ainsi qu'un **Trello** et un groupe **Teams** afin de pouvoir communiquer au maximum l'avancée du projet mais sous l'œil acéré de notre professeur, donc c'est pour ça qu'on a créé un groupe **discord** (sans le prof ducoup ;)).

L'**avancée du projet**, selon le professeur fut **fulgurante** : « Ce que vous avez fait la... c'est ce qu'on appelle un sprint. » (Nous avons pensé à ce moment-là : « dans ce cas-là c'est un marathon sprinté » car il nous restait encore bien du travail à faire). De plus, le projet ayant commencé aux alentours de **début mars** ne nous laissait que peu de temps sans compter les **épreuves du bac de spécialités** qui arrivaient dans les plus brefs délais.

Entre **réunion IRL** pendant les cours de **spé NSI** et **appels discord** le soir ou la nuit pendant le week-end (OUI, Nous ne dormons pas (perte de temps)). La **communication** était à son paroxysme et les idées fusaient dans les têtes de nos jeunes programmeurs, nombre d'entre elles n'ont pu être réalisés par faute de **manque de temps** et de **prise de conscience réaliste** sur le temps restants avant la date butoir à laquelle notre destin allait être scellé.



LES ÉTAPES DU PROJET :

L'idée d'un Tower Defense est venue **naturellement**, en effet, Scotty et Jérôme jouant à des jeux depuis leur enfance sur des **sites internet avec Flash Player** (*R.I.P. Flash Player, tu nous manques*), ils découvrirent un type de jeu qui deviendrait un de leur préféré, le Tower Defense, auquel ils ont été introduits par la **série de jeu Bloons TD**.

Donc lorsque leur professeur leur annonça qu'un **projet** devait être réalisés avec toutes les connaissances acquises durant les **deux années de spécialité NSI**. Avec l'**envie de se dépasser** ainsi que d'**apprendre plus**, l'idée d'un Tower Defense rentrer parfaitement dans ces critères étant donné sa nature de **jeu en temps réel** qui imposait une **maîtrise du nombre d'opérations** réalisés afin de **conserver un nombre d'images par secondes satisfaisant**, et car ils savaient que sur un projet qu'ils aimaient, ils allaient pouvoir se donner à fond et être motivé.

La priorité fut alors de choisir comment réaliser le projet, au vu de la nature de celui-ci, une **interface graphique** était de mise, nous avons donc le choix entre **tkinter** ou **pygame**, mais notre choix se porta vers pygame car **pygame** est plus orienté jeu vidéo et **possède donc beaucoup de fonctions prédéfinies** aidant ceux voulant réaliser un jeu vidéo comme nous. *C'est pas du tout car on a déjà un projet avec tkinter et que ce fut un carnage...* Puis la **P.O.O** est venu naturellement toujours en corrélation avec la nature d'un Tower Défense car pour avoir des ennemies, les classes semblaient bien plus appropriées. Et lors de la **création de plusieurs types d'ennemies et de tours**, l'**héritage** semblait comme **LA solution** à tous nos problèmes et à ce qui aurait pu être des centaines voire des milliers de lignes de codes supplémentaires.

Les **plus gros moments de réflexion** furent comment expliquer à notre professeur que nous n'allions finalement pas participer aux trophées NSI (mais il n'est pas dupe et a vu clair dans notre **poisson d'avril**, *pas fun monsieur*).

> FONCTIONNEMENT ET OPÉRATIONNALITÉ :

Le projet est fini sur le plan initial prévu pour le rendu, en effet, le **gameplay principal est fonctionnel** :

- On peut **acheter des tours**, les **placer**, les **améliorer**, les **vendre**
- On peut **lancer des vagues**
- Les tours tirent sur les ennemis leur infligeant des dégâts et lorsque la vie d'un ennemi tombe à 0, il meurt et donne de l'argent au joueur
- On peut perdre si le nombre de vie du joueur tombe à 0, ce qui peut arriver si des ennemis atteignent l'autre bout de la carte
- On peut gagner si le joueur arrive à la fin des vagues en ayant encore au moins 1 point de vie

Le projet possède **aussi des menus** :

- Un menu principal
- Un menu de sélection de carte
- Un menu Encyclopédie permettant d'avoir des informations sur les tours et ennemis du jeu
- Des sous-menus (sélection de difficulté, confirmation pour quitter, défaite, victoire...)

Du contenu a été ajouté de sorte à avoir une **expérience de jeu satisfaisante** :

- **12 tours uniques** avec 2 chemins de 5 améliorations
- **6 cartes** sur des thèmes différents et des configurations différentes rendant chaque carte unique au vu des possibilités de stratégies qu'elle offre
- **14 types d'ennemis uniques** devant obliger le joueur à s'adapter et diversifiant les surprises réservées au joueur
- **100 vagues différentes** proposant une expérience variée et assez longue franchement 100 vagues c'est super long

La présence de bugs est très limitée et ne gêne pas l'expérience du joueur, en effet, il n'y a aucun bug pouvant mener au plantage du jeu, seul quelques bugs visuels très légers subsistent. Ceci est attesté par nous, ayant lancé à de **TRES NOMBREUSES** reprises le projet **après chaque ajout** pour **tester tout ce qui pourrait mener à un plantage**, ce qui a aussi eu pour effet de nous permettre d'attester que le projet est très simple d'utilisation, en effet si toutes les installations préalables ont été faites (pygame) il suffit d'exécuter le fichier « run.py ».

> OUVERTURE :

Dans l'état actuel, le projet pourrait être **publié sur différentes plateformes** (Steam, Itch.io...) pour **Windows et Linux**. On pourrait **envisager une sortie sur Android** via le Play Store (Il faudrait bien évidemment faire les modifications requises pour le tactile et le reste.) afin de **diffuser le jeu au plus grand nombre** car en effet, le genre du Tower Defense est un genre qui est très adapté au port sur surface mobile (Bloons TD, Kingdom Rush...). Et pourquoi pas **monétiser le jeu** sur Android **grâce au service Google Ads**.

Malgré cela, il est **toujours facilement envisageable d'améliorer le projet**, donc voici une **liste d'idées** que nous avons eu, certaines qui n'ont pu être réalisées par manque de temps, et d'autres qui nous sont venues en y réfléchissant :

- **Ajout de contenu** (nouvelles tours, cartes, ennemis) car en effet, à l'heure actuelle, le projet est selon nous assez complet pour proposer une expérience de jeu satisfaisante mais l'ajout de contenu permettra plus de re jouabilité au jeu qui n'est pas assez présente si on n'est pas un fan du genre Tower Defense.
- **Ajout de récompense** en cas de victoire et d'une boutique pour acheter des pouvoirs à usage unique
- **Ajout d'un tutoriel** si c'est la première fois qu'on lance le jeu, afin de faciliter l'expérience du joueur si c'est la première fois qu'il joue
- **L'ajout de nouveaux modes de jeu** (chemin inversés, limitations à certaines tours...), qui permettra d'ajouter encore une fois de la re jouabilité et plus de défi selon le mode choisi hormis les difficultés déjà présentes.
- **L'ajout de fonctionnalités ergonomiques** pour le joueur (lancement automatique des vagues, bouton pour régler le son au lieu d'un bouton ON/OFF...) pouvant permettre une expérience de jeu plus agréable.
- **Implémentation de plus de langues**, ceci permettra une diffusion plus facile et plus grande du jeu dans les parties du monde ne parlant pas le français.
- **Implémentation d'un système de records** afin de pouvoir comparer ses records avec ses amis et leur montrer à quel point on est meilleur qu'eux .

Si le projet était à refaire, que changeriez-vous dans l'organisation, les fonctionnalités du projet et les choix techniques :

Scotty Pruvost : « Personnellement, si on devait recommencer, je pense qu'une grande majorité du projet restera le même car j'aime bien l'atmosphère du jeu ainsi que l'idée du projet. L'organisation aussi était plus que correcte je trouve, je pense que ce que j'aurais fait différemment aurait été de diviser le fichier principal en plein de plus petit fichier contenant les fonctions utilisées et créer une classe pour le mode et la difficulté afin de faciliter l'implémentation de futures modes de jeu »

Jérôme Vandewalle : « Si tout était à refaire, je ne changerais quasiment rien : en effet, de mon point de vue, le projet correspond tout à fait à mes attentes et je dirais même que je ne m'attendais à aller si loin. Il reste évidemment quelques points qui auraient pu être améliorés comme la propreté du code, ou bien l'organisation pour les documents du projet (les photos et la vidéo auraient pu être de meilleure qualité si nous y avions pensé avant les vacances.) Dans l'ensemble, je suis plus que satisfait du projet et fier d'avoir pu me dépasser pour ce projet »

Mathis Van Uytvanck : « Si tout était à refaire , je changerais le langage de programmation pour le faire en HTML, CSS , JavaScript pour permettre à plus de monde de pouvoir le tester et avoir plus de retours pour l'améliorer plus facilement mais le reste resterait pareil sur la même organisation du travail, le même univers »

DOCUMENTATION

Pour le fonctionnement du jeu, il est impératif de posséder **Python3** ainsi que d'avoir installé au préalable le module **pygame**.

Dans le cas contraire, afin d'installer le module pygame de python, veuillez exécuter la commande suivante à partir du cmd.

```
C:\Users\...> pip install pygame
```

Pour le projet, nous avons utilisé des musiques de **Waterflame** à qui appartient tous les droits des musiques utilisées.

(Lien de sa page **Newgrounds** : <https://waterflame.newgrounds.com/>)

L'entièreté du projet fut réalisée en **langage python** avec une majorité de **programmation orientée objet** (P.O.O) et de la **programmation procédurale** afin de faciliter les interactions qui se déroulent pendant le jeu et augmenter la lisibilité de la structure du programme.

Le **stockage des données** est entièrement local et réalisé grâce à la **structure de données JSON**.

Les **éléments graphiques** sont stockés en tant que fichier sous format **PNG**, les éléments audios, quant à eux, sont au format **MP3** ou au format **OGG**.

Dans le projet, **multiples bibliothèques** ont été utilisées comme **os, json, ctypes, copy, random**.

IMAGES DU JEU

IMAGES LORS DE LA REMISE DU PROJET

Menu principal



Menu de choix de la carte (depuis le menu principal)



Interface lors du début de partie (après la sélection d'une carte et d'une difficulté)



Image lors d'une partie



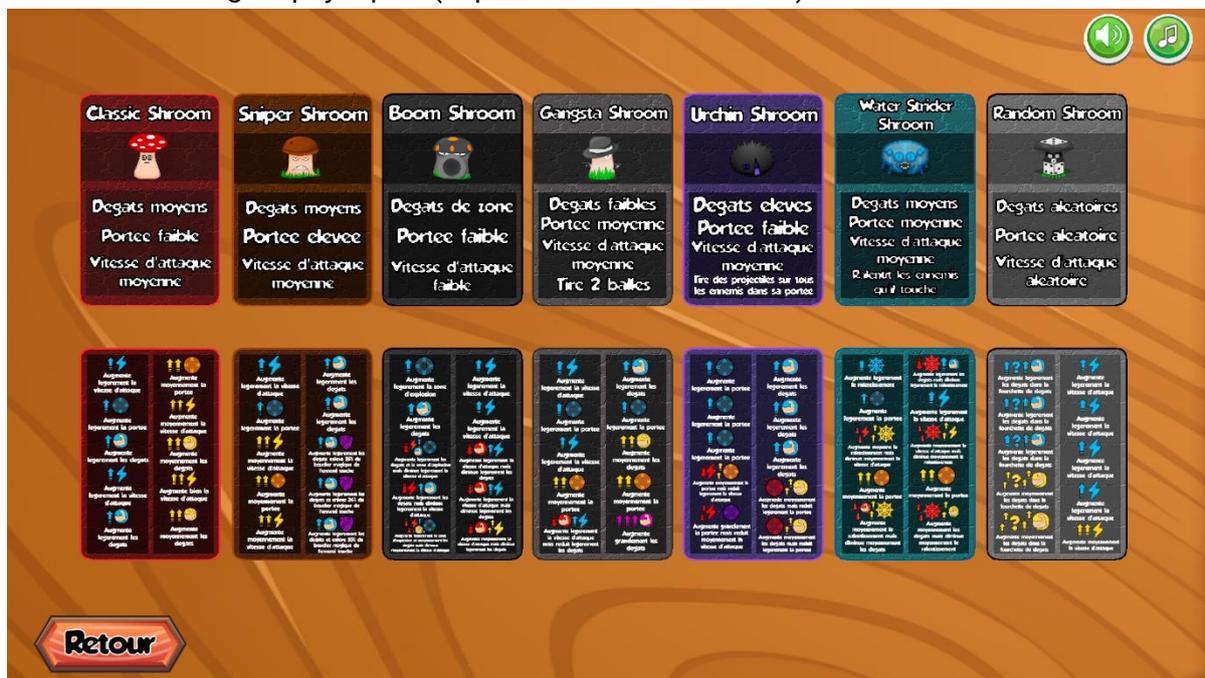
Menu Encyclopédie (depuis le menu principal)



Menu Tours (depuis le menu Encyclopédie)



Menu Tours à dégâts physiques (depuis le sous-menu Tours)



Menu Tours à dégâts magiques (depuis le sous-menu Tours)



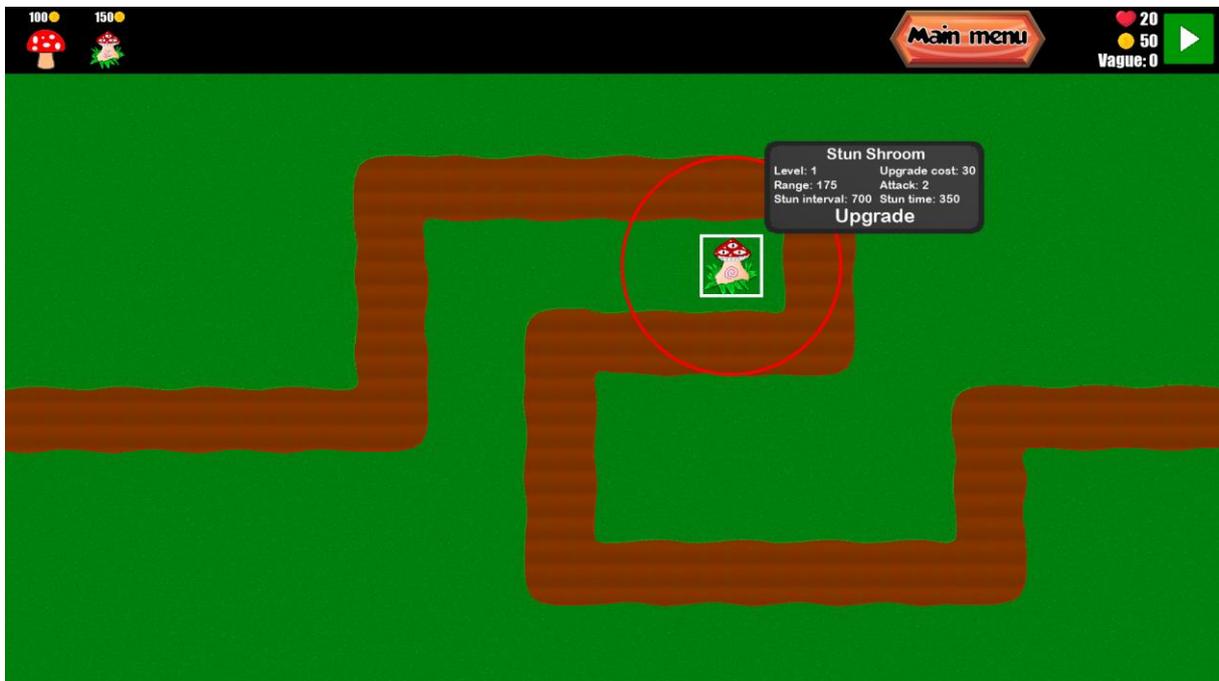
Menu ennemis (depuis le menu Encyclopédie)



IMAGES LORS DU DEVELOPPEMENT DU JEU

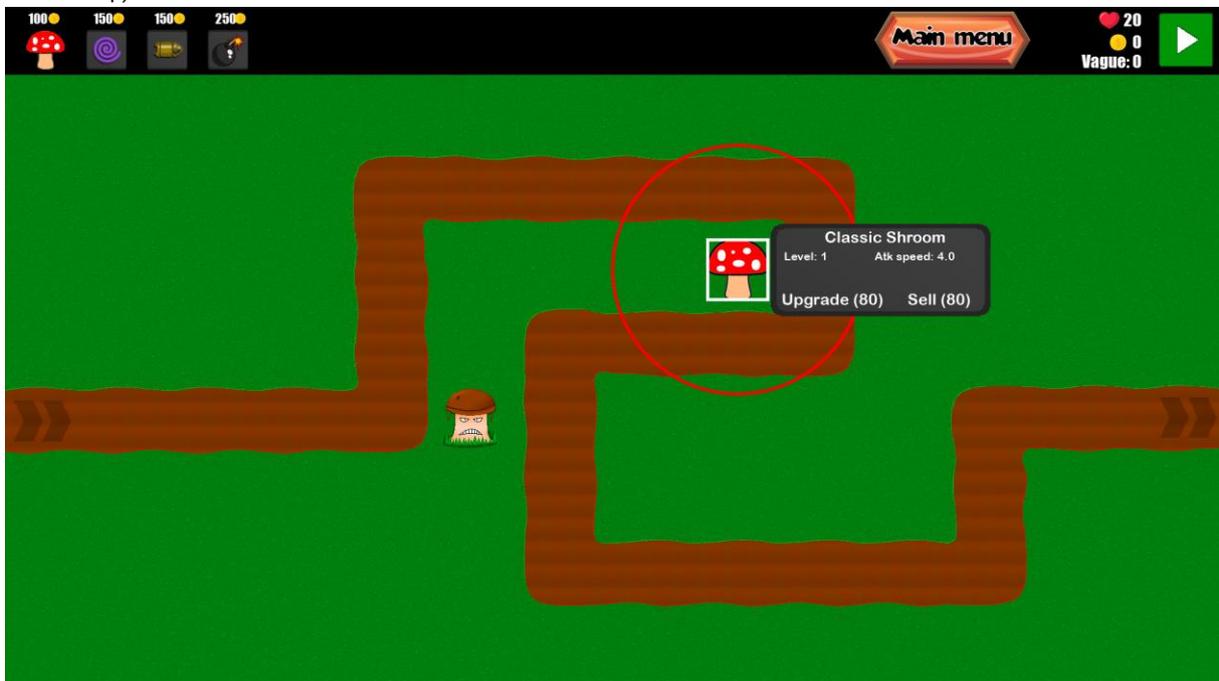
12 mars 2023

(2 tours disponibles, 2 cartes disponibles, affichage de la portée différente, améliorations différentes, pas de menu échap)



15 mars 2023

(Menu principal présent, 4 tours disponibles, pas de tours aquatiques, améliorations différentes, 2 cartes disponibles, pas de menu échap)



29 mars 2023

(Fun fact : la carte du screen n'est plus jouable car le directeur artistique du jeu (Scotty) la trouvait vraiment moche alors que c'est lui-même qui l'avait faite, en plus la map n'était pas intéressante en termes de gameplay (le 1,5 faisait d'ailleurs référence aux nombres d'espèce de champignon dans le monde qui est d'environ 1,5 million (mais quel travail de recherche dans ces cartes))) (8 tours disponibles, 5 cartes disponibles (dont une qui fut retirée), pas de tours aquatiques, système d'améliorations différent)

