SYNAPSE : RAPPORT DE PROJET

Groupe Synapse  
Mehdi Alaoui  
Timaël Andrié  
Mathéo Ernesto Chacon  
André William Cadet  
Jeremias Kuehne

# Introduction

C’est le 2 octobre 2021 que le studio LevelHead sort *Turing Complete* – un jeu de puzzle basé sur les portes logiques. Succès immédiat, le jeu est la parfaite illustration d’un jeu à visée pédagogique qui fonctionne : le jeu est plaisant à jouer et la transmission de savoir est efficace. Mais malgré son succès, *Turing Complete* a un côté austère qui le rend difficilement accessible à certaine populations. Nous avons donc décidé de nous inspirer fortement de Turing Complete pour tenter de proposer une expérience plus adaptée aux plus jeunes. En effet, les portes logiques ayant une place centrale en informatique (Sanchez, 2012, p. 17), il nous semblait pertinent de produire un jeu visant à enseigner leur fonctionnement aux enfants.

Notre équipe est composée de quatre étudiants de l’EPFL ; Mehdi Alaoui,   
Timaël Andrié, Mathéo Ernesto Chacon, et André William Cadet, tous les quatre en Bachelor en informatique, ainsi qu’un·e étudiant·e de l’Unil, Jeremias Kuehne, en Master en informatique pour les sciences humaines. Tous les membres de l’équipe ont à la fois de l’expérience dans la programmation et dans le développement de jeux.

Ce travail présente à notre sens deux principaux éléments forts. En tant que jeu d’abord, il permet de rendre accessible une partie des connaissances transmises par *Turing Complete* à une population qui ne pourrait y avoir accès autrement. En tant que projet ensuite, il a été pour nous une expérience de game design et de développement de jeu dans lequel des problèmes de production ont dû être affrontés.

# Concept de game design

Synapse est un jeux qui présente deux facettes. La première est celle, évoquée plus haut, qui est inspirée de *Turing Complete*. Il s’agit de la partie *puzzle* du jeu. On y sélectionne des portes logiques à poser sur une grille et à relier entre elles afin d’obtenir l’output demandé pour chaque input donné. Le tout est très simple, et c’est ici le feedback qui est au cœur du plaisir de jouer. En effet, tenter de valider un niveau déclenche une petite animation de test qui vérifie que chaque input donné produise le bon output. Cette succession de tests est représentée de manière graphique (comme c’est le cas dans Turing Complete), ce qui vise à donner un sentiment de satisfaction à la personne qui joue. Il existe un puzzle par porte logique (NOT, AND, OR, NAND, NOR, XOR). Ces puzzles sont répartis dans un arbre qui structure la progression de la difficulté en ne rendant certains puzzles accessibles qu’après en avoir réussi d’autres

L’autre facette du jeu est la partie *incrémentale*. Très inspirée de jeux tels que *Satisfactory* ou *Factorio*, cette partie a pour objectif d’être la source de motivation principale pour les personnes jouant au jeu. Le design de cette partie est minimaliste ; il n’existe qu’une seule ressource – l’argent – et cette ressource ne peut être récoltée que d’une manière – au moyen de robots. La seule fonction de cette ressource est d’être dépensée contre des robots ou des améliorations pour ces robots. Les améliorations demandent en outre de réussir l’un des puzzle mentionnés plus hauts, faisant le lien entre les deux facettes du jeu.

* Idée de base
  + Incrémental :
    - Personnage déplaçable
    - Hub : production de robots, dépôt des ressources, achat d’upgrades
    - Robots lvl 0 : ramassés et déposés, collectent puis vidés dans hub
    - Robots lvl 1 : naviguent tout seul
    - Robot lvl + : bougent plus vite, récoltent plus vite, plus grand réservoir
  + Puzzle
    - 1 par porte logique
    - Arbre pour ordre
    - Input avec 1/0, output
    - Possibilité de poser des éléments
    - Possibilité de connecter des éléments
    - « test » final
    - Globalement +/- identique à Turing Complete dans l’interaction
* Dans les faits
  + Pas de partie incrémentale
  + Partie puzzle correspond à ce qui était attendu, avec un manque de polish.
* MOBILISER UNE SOURCE GAME STUDIES/DESIGN

# Gestion de projet et répartition des tâches

La répartition des tâches s’est principalement faite selon les compétences des membres de l’équipe. Les intérêts propre des membre de l’équipe ont également joué un rôle.

|  |  |  |
| --- | --- | --- |
| NOM | COMPÉTENCES & INTÉRÊTS | TÂCHES |
| Mehdi Alaoui | * Idée de pitch préexistante * Expérience préalable en développement de jeu * Maîtrise de la programmation | * Pitch initial (terminé) * Développement - Hub central  (En cours, pas implémenté) * Développement – Environnement  (Pas commencé) * Graphismes - Hub central (Pas commencé) * Graphismes – Environnement (Pas commencé) * Sons - Hub central   (Pas commencé)   * Sons – Environnement (Pas commencé) * UI – Hub central (Tâche effectuée en plus des tâches de base) |
| Timaël Andrié | * Intérêt pour le game design * Expérience préalable en développement de jeu * Maîtrise de la programmation | * Game Design  (Terminé) * Développement - Personnage joueur (En cours, pas implémenté) * Graphismes – Personnage joueur (En cours, pas implémenté) * Sons - Personnage joueur (Pas commencé) |
| Mathéo E. Chacon | * Maîtrise de la programmation * Expérience préalable en développement de jeu | * Développement – Robots (En cours, pas implémenté) * Graphismes – Robots   (En cours, pas implémenté)   * Sons – Robots   (Pas commencé) |
| André W. Cadet | * Intérêt pour le game design * Maîtrise de la programmation * Expérience préalable en développement de jeu | * Game Design (Terminé) * Développement – Puzzles (Terminé) * Graphismes – Puzzles (Terminé) * Sons – Puzzles (Pas commencé) * UI – Puzzles (Tâche effectuée en plus des tâches de base) |
| Jeremias Kuehne | * Compétences en game design * Compétences en programmation * Compétences en rédaction * Expérience préalable avec Godot | * Game Design (Terminé) * Développement – UI (En cours, partiellement implémenté) * Graphismes – UI (En cours, partiellement implémenté) * Sons – UI (En cours, partiellement implémenté) * Rédaction - Game design document (Terminé) * Rédaction - Rapport final (Terminé) |

Problèmes :

* Pas de producteur
* Mais même avec prod., différence de culture ?
* Pas de timeline claire
* Tâches trop vastes, pas assez granuleuses
* Absence de communication

# Concept de gamification

* Reprendre GDD : pas de matériau scientifique, mais des compétences techniques
* Le jeu s’adresse aux enfants
  + Donc design minimaliste / simple
  + Motivation principale : « numbers go up »
  + Vecteur de transmission de savoir : les puzzles
* Mobiliser Nicholson
* Recherche d’une récompense extrinsèque (incrémental) qui amène les joueureusexs à s’engager dans une activité (les puzzles) qui comprènent une récompense intrinsèque (la validation visuelle des puzzle)
  + La récompense extrinsèque ne concerne le puzzle qu’indirectement. Résoudre un puzzle ne fait pas monter le score. On a malgré tout une forme de récompense extrinsèque via les amélioration des robots. Comme il s’agit d’un moyen plutôt que d’une fin, nous espérons que cela n’entraîne pas une frustration face aux puzzles (nbp : surtout que le score est lui-même un moyen d’acheter plus de robots etc. Il n’y a dans le fond que des moyens dans un jeu incrémental, et aucune fin). Mais des playtests seraient nécessaires pour nous en assurer.
  + Notre prototype ne contient que le jeu de puzzle. Donc seulement la motivation intrinsèque. Et ça marche bien.
  + Playtests : révèlent l’efficacité du Gameplay mis en place par André
* Typologie des modes de visualisation vidéoludiques (cf bouquin Selim)
  + Caméra aérienne à translation bidimentionnelle avec activation synchrone avec cartoonisme (incrémental)
    - Sensé permettre une identification au personnage
    - On explore le monde par son avatar
    - Participe au côté hyperbolique et absurde du « numbers go up »
  + Caméra donnant sur un plan fixe avec abstraction symbolique (puzzle)
    - Focus sur lae joueur.euse.x, c’est elui qui fait et non un personnage
* Limites du projet : on pourrait le voir comme un sous Turing Complete. Aussi sa force : un Turing Complete pour enfant plutôt qu’un sous Turing Complete.

# Conclusion

* Un projet qui aurait dû rater mais dont l’aspect compartimenté a permis de contrebalancer les problèmes de production

# Bibliographie