



Dossier de conception

Protocole phronesis

L'ÉQUIPE



Adel Sanaa

Creative Technologist
Project Manager



Jules Mesnil

Creative Technologist



Jason Chevalier

Directeur Artistique
Game Designer
3D Artist



Gaëtan Jestin

Creative Technologist
Directeur Artistique
3D Artist



Emie Pasquier

Graphic Designer

PRÉFACE

Notre projet est né d'un désir de transmission. Nous voulons concevoir quelque chose de durable, une expérience ludique d'exploration pensée comme un héritage pour les promotions futures. L'idée est de laisser une trace physique et de laisser une empreinte de notre passage à Gobelins.

Nous voulons créer une expérience vivante comme un fil rouge entre les générations.

Présentation du projet

Univers

ARGs

Onboarding

PITCH

Phronesis est plus qu'un jeu : c'est un passage, un protocole de test, comme un appel lancé d'une promotion à l'autre.

Chaque année, les étudiants d'ECNI découvrent une machine intrigante qui distribue son premier message. Pour le reste, c'est à eux de le trouver, mais seuls les plus curieux et dégourdis iront jusqu'au bout.



CIBLE

Ce projet a été conçu par et pour les étudiants d'ECNI, en Master 2 à Gobelins. Pour ceux qui passent l'année à jongler entre l'école et l'alternance mais qui gardent encore de la place pour des bêtises.

Ici, nous nous adressons à une promotion un peu hors du cadre qui s'épanouie dans sa curiosité.

SYNTHÈSE RECHERCHE EXPLORATOIRE

L'exploration de la pertinence d'un ARG (Alternate Reality Game) immersif au sein des Gobelins s'est révélée très positive.

Nous avons donc construit à partir d'une fiction imaginée comme un projet oublié des années 80, un dispositif mêlant énigmes, pédagogie, design interactif et narration.

Axes explorés

L'attrait des étudiants pour les expériences immersives et les énigmes basées sur le déchiffrage.

La motivation à s'engager dans des épreuves ludiques, non obligatoires mais intrigantes.

Les attentes des étudiants et leurs formats d'interactions préférés (numérique, tangible, collaboratif...).

Méthodologie

5 entretiens semi-directifs avec des étudiants de profils variés (DA, Interactivité, Développement).

Observations croisées des comportements lors de projets immersifs ou de workshops précédents.

INTERVIEWS

“Ouais je m’investis, surtout pour des trucs qui sont liés à la formation mais par contre qu’à son début.” —

Ariste Gautier

“Un ARG, s’il est bien fait, c’est pas juste un jeu. C’est une façon de hacker le réel. Et dans une école comme Gobelins, c’est un terrain de jeu parfait.” —

Stan Husson

“On a envie d’appartenir à quelque chose d’un peu secret, un collectif dans l’école, un truc que seuls certains voient.” —

Camille Ngo

“Pourquoi pas découvrir des choses que j’ai pas l’habitude de faire. J’aurais plus envie de jouer avec des ateliers.” —

Amandine Quelled

PROPOSITION

Nous avons imaginé un projet à la frontière entre fiction, pédagogie et rituel de promotion.

Notre proposition est de créer une machine physique accompagnée d'une narration réactivée chaque année, qui donne accès à un ARG (Alternate Reality Game) décomposé sur quatre périodes. Pensé comme un protocole secret légué d'une promotion à l'autre, ce jeu immerge les étudiants dans une enquête mêlant énigmes, narration transmédia, interactions tangibles et défis techniques.

Au travers de cette expérience :

Les étudiants s'affronteront en coopération pour résoudre des modules narratifs et interactifs.

Ils valideront leurs découvertes via la machine, qui centralise les scores et imprime les instructions de chaque période.

L'ensemble du jeu est gamifié pour encourager l'engagement, mais reste volontaire et parallèle au programme officiel.

Le projet repose sur la fiction d'un groupe d'étudiants des années 80 ayant découvert un secret sur leurs intervenants. Trop visibles à l'époque, ils ont été réduits au silence mais ont laissé derrière eux un test :

Phronesis.



Présentation du projet

Univers

ARGs

Machine

Son & pédagogie

UNIVERS NARRATIF

L'univers narratif du projet s'ancre dans une fiction simple. En 1987, un groupe d'étudiants des Gobelins a découvert **WUJUTU** sur les intervenants encadrant la formation. Ils auraient tenté de partager leurs découvertes, mais leurs efforts ont été immédiatement étouffés, les réduisant à néant.

C'était sans compter qu'avant de disparaître, ils laissèrent derrière eux les bases d'un protocole. Celui-ci est destiné à aider les futures promotions à continuer à lutter contre la réduction au silence par les intervenants.

Notre jeu s'inspire de cette base et de ces valeurs. L'idée n'est pas de rendre cette fiction spectaculaire mais de créer un cadre narratif cohérent, engageant et suffisamment ouvert pour évoluer avec les années.

Dans l'optique de renforcer leur défense contre la société secrète orchestrée par Veronique Ficara depuis la Grèce, les étudiants en ont monté une sous le nom de Phronesis.

NOM DU PROJET

Phronesis, du grec φρόνησις /*frɔ̃ˈniːsis*/ est un concept philosophique utilisé depuis l'Antiquité. Ce terme peut s'expliquer aujourd'hui comme une forme de sagesse appliquée à l'action humaine. Cela implique la capacité d'agir correctement dans une situation concrète ainsi que d'agir avec discernement.

Selon Aristote, la phronesis s'ancre dans une raison pratique où l'expérience, le jugement moral et le caractère sont des éléments clés. Pour lui, il ne s'agit pas d'appliquer des règles universelles mais de délibérer avec justesse en fonction des circonstances.

Enfin, selon Platon, la phronesis est une forme de compréhension morale essentielle. C'est un élément immuable à la vertu.

De cette façon, phronesis décrit parfaitement ce qui est attendu des participants : prendre les bonnes décisions quand celles-ci sont importantes.



phronesis™

TON ET LANGAGE

Le ton de Phronesis oscille entre le sérieux intrigant et la connivence étudiante. Nous ne parlons pas “au nom d’un jeu”, mais comme une voix transmise d’une promo à l’autre : confidentielle, complice, parfois légèrement solennelle, mais toujours accessible.

Ici, nous nous adressons à des étudiants, pas à des joueurs experts. L’idée est de communiquer avec un air mystérieux et des clins d’œil bien placés.

Nous avons choisi un langage clair mais codifié, chaque message imprimé par la machine peut être pris au sérieux, ou interprété autrement.

On entretient également une ambiance de rituel étrange sans verser dans le dramatique.

Nous avons voulu communiquer d’égal à égal avec les étudiants des années futures. Nous ne nous positionnons pas comme des maîtres du jeu ou des enseignants mais plutôt comme des élèves qui veulent aider leurs camarades.

ONBOARDING

La réussite de notre onboarding* est une étape cruciale pour attraper le plus de joueurs dès le début de l'expérience. L'idée est de capter l'attention de ceux qui, d'eux mêmes, ne seraient pas allés voir la machine.

Il marque l'entrée dans l'univers du jeu de façon subtile. Il ne révèle ni la profondeur du système, ni le nombre total d'étapes. Ainsi, nous avons pensé à utiliser comme canal la vidéo, diffusée directement par vidéo-projecteur aux élèves. Par manque de temps, l'onboarding de la première année se fera par mail.

Le mail présentera la structure type du contenu et sera une accroche intrigante contextualisant le projet.

*« L'onboarding » désigne le processus d'accueil et d'accompagnement qui permet au joueur de comprendre, adopter et s'intégrer efficacement à un système, un produit ou un univers.



Présentation du projet
Univers
ARGs
Machine
Son & pédagogie
Impact & viabilité

ALTERNATE REALITY GAME

L'ARG est le cœur du projet, c'est une expérience immersive unique conçue, non pas comme un jeu au sens classique, mais plutôt comme un système d'énigmes, de récits et d'interactions physiques qui s'intègre discrètement dans leur quotidien. La machine en est le point d'entrée, mais tout peut devenir indice : document, objet, espace partagé.

Il se déroule sur les quatre premières périodes en présentiel à Gobelins, afin de n'empiéter ni sur l'alternance, ni sur le projet de fin d'année de M2. Pensé comme un rituel de promotion, l'ARG revient chaque année, conservant une structure stable et rejouable.



SYNTHÈSE RECHERCHE COMPÉTITIVE

Cicada 3301

2012

CipherHunt

2016

A Labyrinth

2020

Neurocracy

2021



ARG organisé sur Internet et mettant en jeu à titre principal des compétences en cryptographie et en informatique. Une nouvelle série de défis a été lancée chaque année autour du 5 janvier en 2012, 2013, 2014 et 2016, dans le but affiché de recruter « des individus très intelligents ». L'identité des personnes ou organisations qui ont lancé ces défis demeure inconnue.



The Cipher Hunt était un jeu de réalité alternative et une chasse au trésor internationale créé par le scénariste et doubleur Alex Hirsch, basé sur sa série animée Gravity Falls.



A Labyrinth est un jeu en réalité alternative (ARG) développé à l'Université de Chicago pendant la pandémie de COVID-19. Il a été créé pour stimuler la créativité et créer une communauté en ligne dans le contexte inédit et émergent de la pandémie.



Neurocracy a été décrit à plusieurs reprises comme un jeu en réalité alternative et une fiction hypertexte. En naviguant via leur propre navigateur web, les joueurs parcourent Omnipedia, une encyclopédie en ligne fictive largement inspirée de Wikipédia, pour résoudre l'assassinat de son principal investisseur, Xu Shaoyong.

SYNTHÈSE RECHERCHE COMPÉTITIVE

Immersion

Narration transmédia : utilisation de multiples plateformes (sites web, vidéos, appels téléphoniques) pour disséminer des indices et plonger les joueurs dans une enquête immersive.

Phygital

Intégration dans le monde réel : des éléments du jeu sont dissimulés dans des lieux physiques réels, incitant les joueurs à interagir avec leur environnement pour progresser.

Stéganographie

Utilisation de technologies courantes : exploitation de technologies accessibles (QR codes, fichiers audio modifiés, coordonnées GPS et sessions informatiques personnalisées) pour dissimuler des indices.

Jeu vivant

Évolution du jeu : le jeu évolue au fil du temps, avec des mises à jour régulières et des changements dans l'environnement, maintenant l'engagement des joueurs sur une période prolongée.

GAMIFICATION

Les ARGs, sont fascinants mais souvent réservés à une niche d'initiés. Ceux-ci aiment fouiller des forums, décoder des énigmes en hexadécimal et perdre trois nuits sur une image floue trouvée dans un recoin d'internet. Notre cible étant définie, nous devons élargir ces concepts afin de toucher le plus d'étudiants.

Alors nous avons fait le choix de gamifier l'expérience. L'objectif n'était pas de la simplifier, mais de la rendre plus accueillante, plus progressive, et surtout plus gratifiante. Nous y avons inclus des mécaniques de score, des checkpoints ainsi qu'une compétition assumée entre étudiants. Juste ce qu'il faut pour accrocher dès les premières minutes et garder l'envie d'avancer même quand un point bloquant montre son nez.

Il a aussi été conçu avec les compétences de Gobelins en tête, il n'est ni pour les designers, ni pour les développeurs. C'est un heureux mélange de chaque discipline, toujours avec la coopération en tête. Les étudiants devront, si besoin, s'entraider pour avancer.

GAMEFLOW [Accéder au Gameflow ↗](#)

The screenshot shows a Miro collaborative workspace with the following elements:

- Top Bar:** Browser tabs for "Titled", "Recherches", "DOC D'INTENTION", and "DOC D'INTENTION - DOC". A toolbar on the right includes a profile icon, a share icon, a timer at "03:00", and a "Share" button.
- Left Panel:** A vertical toolbar with various drawing tools like a hand, eraser, highlighter, and shapes.
- Main Canvas:**
 - Période 1:** A flowchart with a central node containing "M" and "A", branching into several paths of colored boxes (purple, pink, yellow, green).
 - Période 2:** A more complex flowchart with a central circular node containing "M", surrounded by multiple levels of branching boxes. A note "Bloc ciment" is at the bottom right.
 - Période 3:** A simple flowchart with a central text box "Validé par Mathieu, bravo à nous 🍌" and two paths leading to green boxes.
 - Période 4:** A large, intricate flowchart with many nodes and paths, including a large orange box at the bottom with the text "Once compte à rebours de 5 secondes".
- Right Panel:**
 - Text Notes:**
 - "joueur, penser à des interactions qu'on attend pas des interactions présentées comme étant celles 'de base'. Exemple : laisser une carte scannée pendant 10s au lieu de vite scanner débloque un truc ; Ou comprendre qu'un message s'affiche quand on coupe le courant de la machine"
 - "→ avoir une grosse mécanique qui restreint le joueur (et en souffre) puis réussir à lui faire réaliser au moment voulu comment il peut se débarrasser de cette contrainte"
 - "Code barre franprix"
 - "Imprimante à indices"
 - "Code polybe ou code de téléphone à ancienne morse"
 - "mots croisés"
 - "effique"
 - "mathématique (résoudre un calcul dur sa mère)"
 - "code des francs maçons"
 - Image Grid:** A grid of small images at the bottom left, showing various scenes and objects.

Présentation du projet

Univers

ARGs

Machine

Son & pédagogie

Impact & viabilité

Bilan

POURQUOI UNE MACHINE

La machine est le cœur du projet. Elle initie le jeu, imprime les instructions, enregistre les clés et garde le score. Elle agit comme un témoin silencieux de l'expérience, mais aussi comme le point de contact physique entre les joueurs et l'univers qu'ils explorent.

Nous l'avons conçue pour rester d'années en années à l'école, pour qu'elle devienne un rituel pour chaque promotion.

Sa construction mêle composants récents et techniques détournés, dans un esprit années 80 assumé. Elle est autant outil de jeu que support narratif.

COMPOSANTS INTERNES

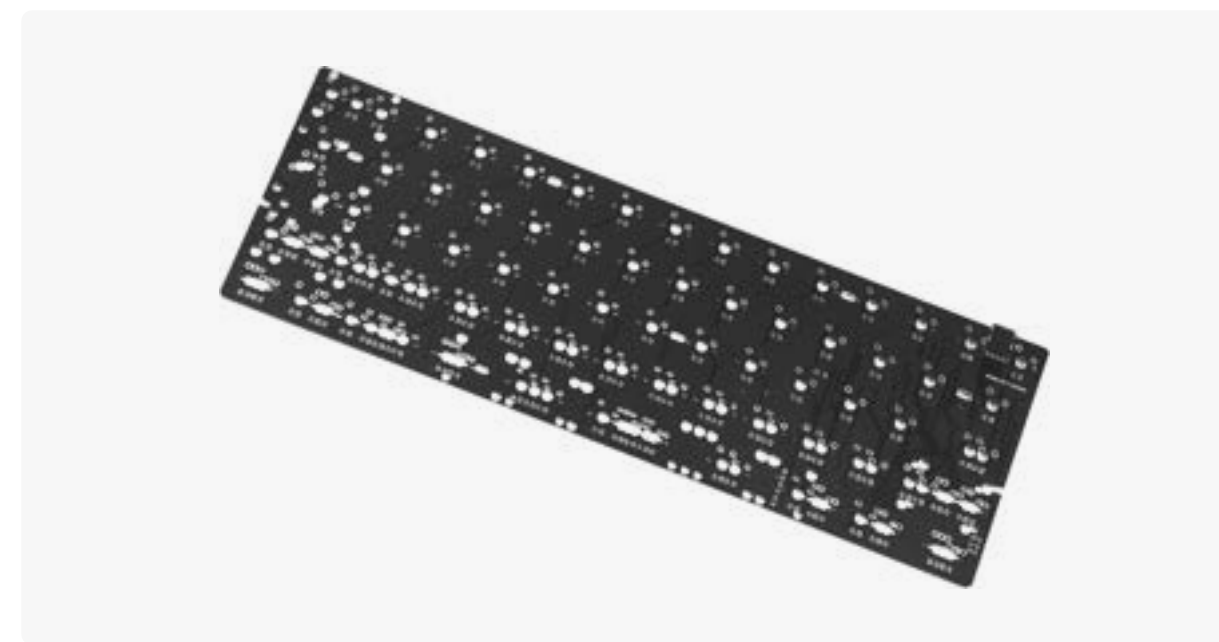
Raspberry Pi 5



Écran e-ink



PCB Clavier



Enceintes



Imprimante thermique



Arduino



Capteur RFID



CONCEPTION

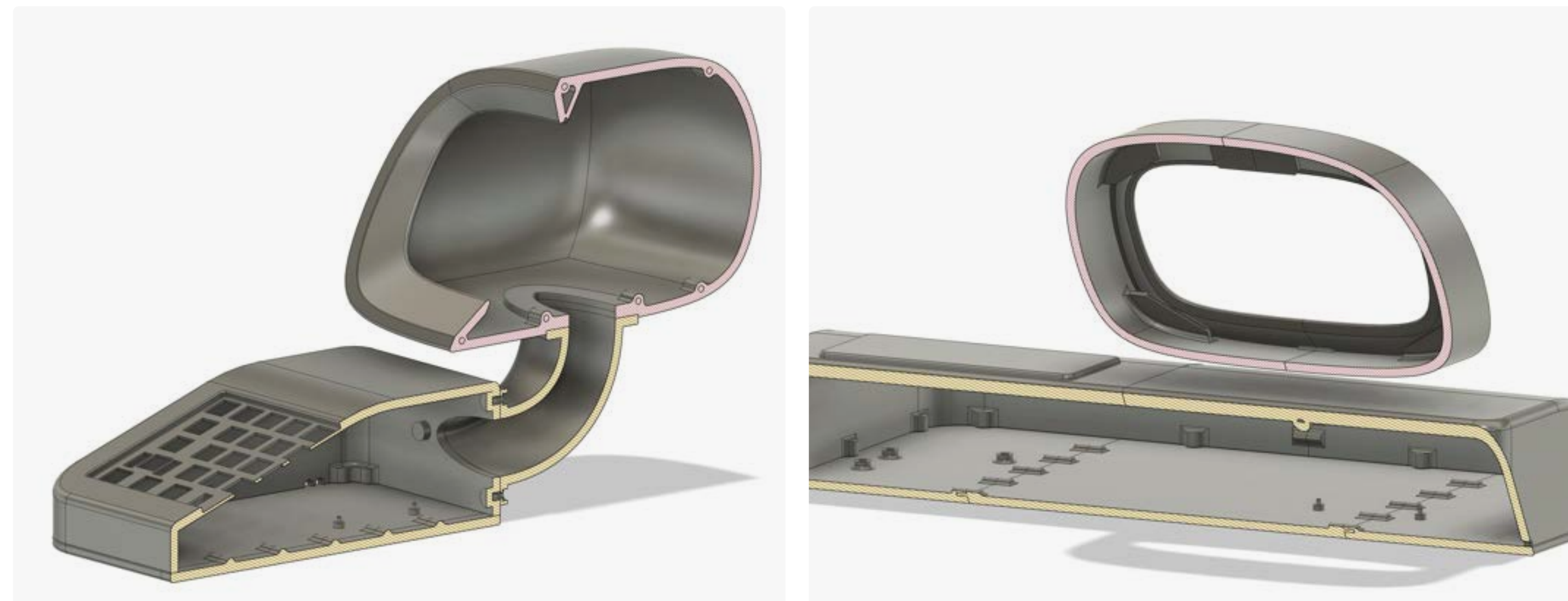
Le design de la machine a commencé sur Blender pour poser rapidement des idées et des concepts en 3D. À cette étape, le but n'était pas la précision technique, mais de tester des proportions, tout en gardant à l'esprit les dimensions réelles de chaque composant. C'était une maquette d'intention, presque comme un croquis en 3D. Ce choix nous a permis d'itérer rapidement ce qui était primordial à ce stade du projet.

Une fois cette première phase validée, on a basculé sur Fusion 360 pour concevoir une version techniquement exploitable. Nous avons pris les mesures précises de tous les composants, le Raspberry Pi, l'écran e-ink, l'imprimante thermique, le clavier, etc. Et nous avons recommencé la modélisation, cette fois-ci de manière paramétrique propre aux logiciels CAD.

Ce changement de paradigme lié à l'utilisation de Fusion 360 nous a amené à une vraie montée en compétence, logiciel que nous avons dû apprendre à utiliser en autonomie, en parallèle du reste du projet.

Ce processus a été long, parfois un peu frustrant (il a fallu recommencer quelques fois), mais essentiel. Il nous a permis de créer une structure stable et prête à être imprimée en 3D.

Les fichiers peuvent également être facilement modifiés et adaptés dans les années à venir. La machine a été pensée pour durer, se transmettre, et vivre plusieurs itérations sans repartir de zéro à chaque fois.



TECHNIQUE

Le projet présentait de nombreux enjeux techniques qu'il a fallu résoudre. Cependant, l'équipe de développeurs voulait appuyer aujourd'hui sur le plaisir qu'ils ont ressenti à se saisir de ces enjeux et à en tirer de nouvelles compétences

Aussi bien qu'en hardware qu'en programmation des logiques, le projet soulevait beaucoup de problématiques passionnantes à travailler.

LES PROBLÉMATIQUES TECHNIQUES

Notre premier réflexe crucial qui nous a saisi était qu'en tenant compte de notre concept, comment assurer le bon fonctionnement d'une machine qui doit être en marche à des périodes précises, à la disponibilité de tous tout en étant résiliente et maintenable ?

Alors, nous avons segmenté notre travail en trois grandes parties distinctes :

La mise en place de l'architecture technique et de la logique principale.

La gestion des données.

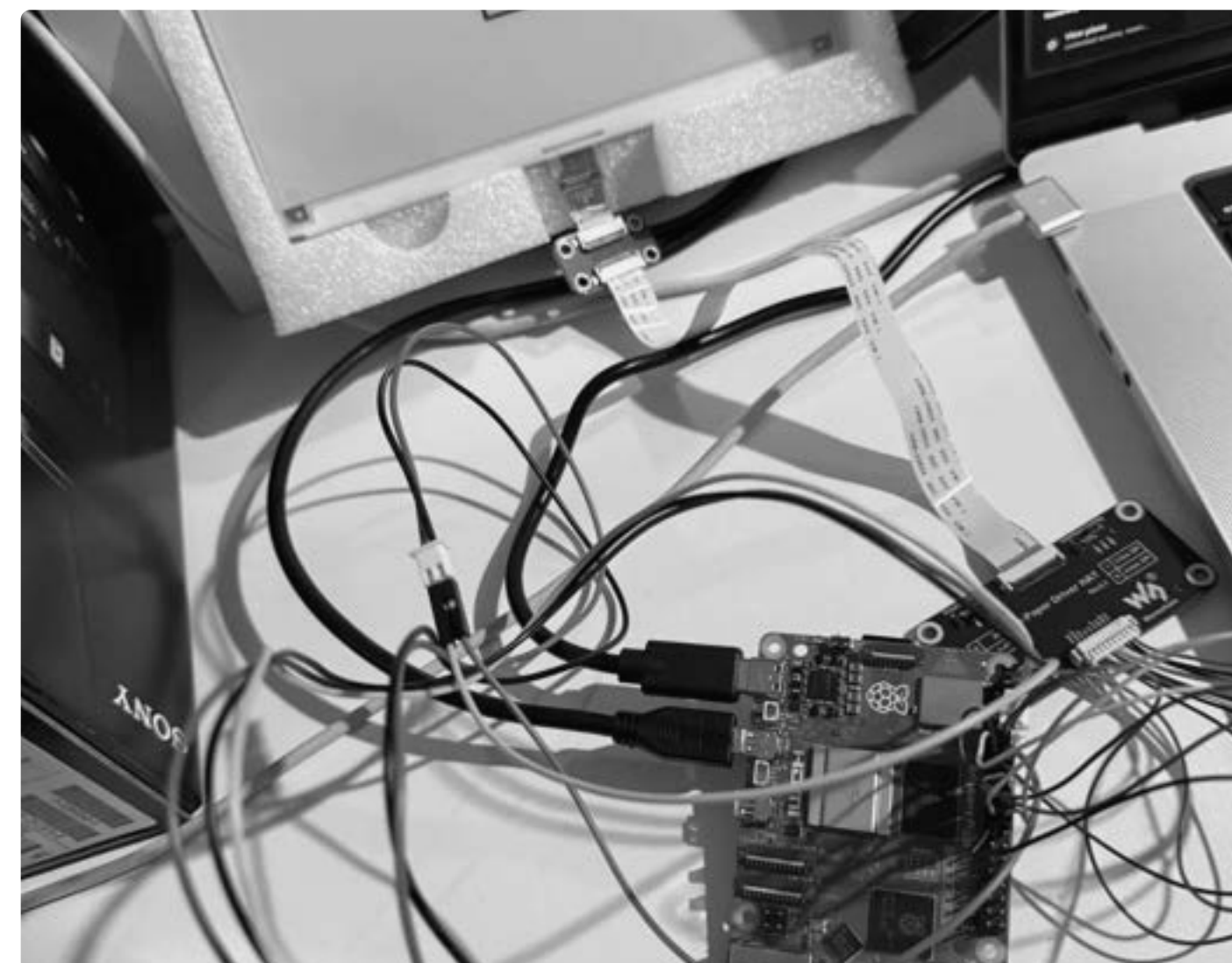
Le monitoring.

RECHERCHE & DÉVELOPPEMENT

Nos priorités étaient en place, il fallait maintenant commencer la recherche et confirmer la faisabilité des grosses parties du projet. Les premières recherches que nous faisons étaient principalement sur la résolution du principal problème :
Sommes-nous capables d'écrire un comportement de machine qui s'active à des temps précis et exécute des programmes qui lui sont configurés au préalable ?

Nous avons aussi beaucoup fait de recherche technique sur la puissance de calcul qui nous fallait. Ce qui nous a orienté vers l'utilisation d'un Raspberry Pi. Nous avons pris la dernière version qui consomme très peu d'énergie, ne dépend pas d'un utilisateur connecté et peut redémarrer automatiquement en cas de coupure.

Nous avons aussi fait énormément de recherches sur l'architecture de façon plus globale qui allait tourner sur le Raspberry Pi. Comment appeler des scripts depuis d'autres, la structure des fichiers, la gestion des événements et la communication entre les logiques, le monitoring, l'écoute d'événements spécifiques (comme le clavier, l'arduino), etc.
Chacun des composants, une fois commandés, ont occupé un temps de recherche et de développement pour déterminer, en plus des documentations constructeurs, ce qui était possible pour nous d'exploiter comme fonctionnements.



L ' ARCHITECTURE

Nous avons opté pour un monorepo pnpm.
Cela permet la mutualisation et réutilisation dans une logique de modules, tout en permettant une compilation des librairies personnalisés de façon individuelles.

Notre repo est scalable, maintenable et surtout compréhensible par n'importe quel développeur qui travaillerait sur le projet.

Chaque grosse logique est indépendante : le core, l'ui, les services et les packages (database, keyboard, events, sounds).

Les langages utilisés sont le typescript et le python.

Les serveurs internes communiquent entre eux via un socket UNIX.

LE CORE

Le core est le programme principal. C'est celui qui est démarré en premier lors de la mise en tension du Raspberry Pi.

Il gère le lancement des serveurs principaux comme les services et l'ui. Nous définissons **les services** comme étant les serveurs qui permettent aux énigmes de fonctionner.

Sa principale mission est de monitorer l'ensemble des services lancés par le core ainsi que les problématiques qui peuvent surgir (alimentation hardware, manque de papier dans l'imprimante, Wifi absent, ...).

Il s'occupe également de mettre en place le socket et de transmettre des événements à différents services (exemple : arduino qui souhaite communiquer à l'UI).

La plupart des services sont externes, hébergés ailleurs que sur le Raspberry Pi. Certains peuvent être internes dans les cas où les énigmes nécessitent d'être joués à l'école, grâce au réseau Gobelins.

Le core est codé en TypeScript.

L'UI

PARTIE 1

L'UI (User Interface) est le service de base démarré par le core. Sa seule mission est de s'occuper de manière indépendante de tout le fonctionnement de l'affichage d'interfaces sur l'écran e-ink.

Son serveur a été écrit en python pour faciliter la communication entre le code et l'écran. Le constructeur de l'écran, Waveshare, nous fournit une librairie open-source codée en C et en Python pour communiquer avec l'écran.

Le principal challenge autour de l'UI, aussi surprenant que cela puisse paraître, a été d'afficher correctement les informations sur l'écran. La technologie est assez particulière, la librairie est pensée pour ressembler au fonctionnement du CanvasAPI en web. On dessine des formes ou des textes à des coordonnées précises de l'écran. Ni plus, ni moins.

Le grand problème, est le temps de rafraîchissement de l'écran. Pour changer les informations à l'écran, ça ne fonctionne pas comme un écran LCD, qui accepte les commandes d'une carte graphique pour changer la couleurs de ses pixels à la volée.

Non, ici il faut créer une image, puis ordonner à l'écran de l'afficher. C'est ce qu'on appelle « afficher l'image avec une initialisation complète ». Ce procédé dure 5 secondes.

Heureusement pour nous, il existe d'autres procédés.

Les procédés d'affichage d'informations dynamiques a été le cœur de nos préoccupations.

Travailler avec l'écran e-ink n'a pas été de tout repos. Imaginez avoir un smartphone qui mettrait 5 secondes à chaque action que vous faites. Allumer l'écran, 5 secondes. Se rendre sur l'écran d'accueil, 5 secondes. Écrire une lettre, 5 secondes. L'effacer, 5 secondes.

Ce n'est pas vivable. C'est pourquoi nous misions sur la deuxième fonctionnalité de cet écran : le rafraîchissement partiel. Ce concept est simple, on définit une zone de l'écran à rafraîchir de façon partielle, on lui envoie l'image qu'on veut y dessiner, puis hop ! Elle se dessine en 400 millisecondes.

On pourrait se demander, alors pourquoi ne pas simplement utiliser ce mode de rafraîchissement ? La réponse est simple, il y existe deux raisons à cela. Tout d'abord, il faut préserver la santé de l'écran. De nombreux rafraîchissement partielle risque de le déstabiliser et à terme, de l'endommager physiquement.

De plus, la technologie e-ink est sujette au ghosting. C'est un phénomène qui arrive lorsqu'on itère trop d'images partiellement qui finissent par laisser une trace visible sur l'écran, même si l'on efface l'image et qu'on redessine par dessus. Lorsque cela arrive, on doit procéder à un nettoyage complet de cinq secondes.

L'UI

PARTIE 3

L'UI en elle-même constitue alors sa propre logique indépendante qui est écrite en Python et communique avec la base de données et est branchée au socket principal.

Cela lui permet d'accueillir des événements depuis le core, d'y répondre.

C'est par ailleurs comme cela que fonctionne notre système de détection du clavier, qui est initié par l'UI qui ensuite communique à l'ensemble du système via le socket les touches qui sont tapées.

LE MONITORING

Le core s'occupe de monitorer ce qui se passe sur les services de l'ARG. Ces logiques sont faillibles, elles peuvent arrêter de fonctionner pour énormément de raisons, en dehors de notre volonté.

Une intervention humaine est alors à prévoir quand il n'y a plus de solutions. Alors, pour prévenir les pannes et autres problèmes pouvant perturber l'expérience des joueurs, nous avons mis en place une multitude de logiques.

Nous avons opté pour balenaOS comme système d'exploitation de notre Raspberry. Fonctionnant sur une base de Linux, il permet la connexion à distance et le monitoring est simple et fonctionne très bien. Le gros point faire c'est qu'il fonctionne par conteneurisation et il est facile d'y effectuer des mises à jour à distance et de connaître l'état du Raspberry Pi.

De plus, nous avons fixé un ventilateur sur le processeur du Raspberry Pi pour prévenir les ralentissements et la surchauffe.

L'IMPRIMANTE THERMIQUE ET LE CLAVIER

L'imprimante thermique est simple d'utilisation et apporte une grande immersion pour les joueurs, qui obtiennent une trace physique du point de départ de leur énigme, ou même de leur comptabilité du score.

Le constructeur recommande également de communiquer à l'imprimante thermique via Python, avec des bibliothèques adaptées. Nous avons alors juste à préciser le message et à l'envoyer via communication serial, en TTL.

Le clavier est, lui, branché en USB-C. Il est personnalisable et personnalisé, aux couleurs de la direction artistique.

Réagir aux touches tapées sur le clavier sans qu'elle ne fasse réagir l'OS du Raspberry, ça a été une bonne recherche. Nous avons fini par décider de partir sur balenaOS, ce qui retirait cette problématique où l'utilisateur pouvait s'interfacer avec la machine sans autorisation.

Cependant, il a fallu tout de même écrire notre propre logique de détection de touches et de raccourcis. Alors, le code écoute les entrées des signaux envoyés par le clavier via son port USB et transmet au système si l'utilisateur a tapé une touche, ou un raccourci (MENU + C par exemple).

Tout ça en TypeScript.

LES DONNÉES

PARTIE 1

Les données représentent le nerf de la guerre de notre machine. Nous avons tout d'abord commencé par les distinguer.

Nous avons deux types de données : les données des utilisateurs et les données de configuration de jeu.

Les données des joueurs,

sont stockés sur une base de données SQLite. Nous l'avons configurée de sorte à ce qu'elle ne puisse pas se corrompre si, par malheur, la machine venait à être mise hors tension involontairement (panne, coupure d'électricité, ...).

Son réflexe, lorsqu'on lui ordonne une écriture, est d'abord de consigner ses instructions dans un journal, puis d'écrire la base de données. Alors, au démarrage de la machine et à période régulière (lors de prochaines écritures), la logique de la base de données vérifie d'abord qu'elle n'a rien de consignée dans son journal, vérifie l'intégrité de la base et reprend là où elle s'était arrêtée.

Nous stockons les informations suivantes : le nom, le prénom, le score, les clés entrées (et quand), la description personnalisée des joueurs dans le Hall of Fame.

Les données de configuration,

sont inscrites de différentes façons. La configuration des périodes et des services dont elles dépendent sont indiqués dans des fichiers YAML. Le core s'occupe alors de vérifier régulièrement la période sur laquelle elle se trouve et de lancer les services qui correspondent.

Aussi, la configuration des clés se fait au préalable, pour défendre le système de la triche. Aucune clé n'est stockée en clair dans le système, nous les cryptons au préalable. Alors, lors de l'entrée d'une clé, le système les crypte, puis les compare à celles qui sont stockées.

Résilience, encore.

Les données sont cruciales car elles stockent toutes les informations relatives au bon déroulement du jeu pour nos joueurs. Alors, nous avons également prévu une sauvegarde régulière de la base de données sur un serveur distant. Le core s'occupe, lorsque c'est possible et régulièrement d'y envoyer une sauvegarde datée.

En cas de problème, une intervention humaine de notre équipe pourra rétablir le jeu correctement à une version proche de l'incident.

LES DONNÉES

PARTIE 3

Le consentement relatif aux données personnelles est important. Nous travaillons en étroite collaboration avec notre chef de formation pour nous assurer de recueillir le consentement et le non-consentement des élèves de chaque promotion.

Le choix d'une base de données SQLite s'est révélée parfaite pour notre utilisation simple. SQLite nous permet d'avoir une base robuste, simple d'utilisation et locale.

Chaque année, le jeu sera mis à jour avec les données des nouveaux joueurs par notre chef de formation, ou par des membres de notre équipe sur demande du dit chef de formation.

Univers

ARGs

La machine

Son & pédagogie

Impact & viabilité

Bilan

Notes de fin

SON

L'absence de taux de rafraichissement de l'écran e-ink impose une contrainte technique importante. Elle empêche animations fluides et transitions instantanées. Pour compenser cette latence visuelle, nous avons intégré un système de feedback sonore simple mais cohérent.

PÉDAGOGIE

Un aspect important de notre projet à été d'inclure un axe pédagogique. Nous avons décidé en accord avec Mathieu de transformer la troisième période de l'ARG en un "Projet Arduino".

Les étudiants auront comme objectif de réaliser un scanner RFID et une plaque imprimée en 3D qui leur permettra de débloquent l'accès à la période 4 à leur retour.

Les lead technique devront commencer par se familiariser avec un Raspberry Pi et un Arduino, puis continuer avec le scanner RFID et les cartes encodées. Ils devront ensuite apprendre à s'interfacer sur la machine et ses contraintes techniques actuelles, pour envoyer la data vers le Raspberry Pi.

Les lead design quand à eux, devront apprendre les rudiments de la modélisation sur Fusion 360, ainsi qu'à lire et comprendre un schéma technique d'usinage. Leur but sera de modéliser une pièce aux dimensions données qui pourra s'attacher à la machine existante, le tout incluant le scanner RFID.

ARGs

Machine

Son & pédagogie

Impact & viabilité

Bilan

Notes de fin

INTENTIONS D'INCLUSION

Le projet n'est pas encore accessible aux personnes en situation de handicap et nous l'assumons. Il a été conçu pour permettre une intégration de l'accessibilité à l'avenir, mais faute de temps, cela n'a pu être intégré dans le calendrier de production de cette première version.

Cependant, nos tests utilisateurs réalisés sur différents profils d'étudiants (timides, try harders, groupes, etc...) nous ont permis de créer un ARG capable de répondre aux attentes du plus grand nombre.

INTENTIONS DE VIABILITÉ

Le projet est pensé pour être vendu à d'autres écoles : clé-en-main ou personnalisé (facturable en plus). Des contrats de maintenance peuvent être envisagés pour assurer le suivi et les évolutions du jeu.

Il a aussi pour objectif d'être pérenne, il restera dans l'école et sera réutilisé chaque année.

L'approche est modulaire - Chaque composant du test est pensé pour pouvoir être ajusté, repensé, remplacé.

Machine

Son & pédagogie

Impact & viabilité

Bilan

Notes de fin

BILAN BUDGÉTAIRE

[Accéder à l'organigramme ↗](#)

BILAN D'ORGANISATION

Le développement de Phronesis a été jalonné de défis techniques, logistiques et humains. Certains ont été anticipés, d'autres moins. Ce projet, qui mêle hardware, narration et expérience ludique, nous a confronté à des contraintes complexes et à notre propre capacité à rester concentrés sur une longue durée.

Pour l'équilibre narratif et la gamification, concevoir un ARG est déjà un défi en soi. Mais vouloir le gamifier, tout en gardant une narration crédible, cohérente et immersive, a complexifié les choix de conception. Nous avons passé beaucoup de temps à ajuster le scoring, les checkpoints et les systèmes de feedbacks pour qu'ils s'intègrent naturellement à l'univers sans casser l'immersion.

Il a aussi été conçu avec les compétences de Gobelins en tête, il n'est ni pour les designers, ni pour les développeurs. C'est un heureux mélange de chaque discipline, toujours avec la coopération en tête. Les étudiants devront, si besoin, s'entraider pour avancer.

BILAN D'ORGANISATION

Concernant le cadre de travail et la dynamique d'équipe, nous avons démarré la production assez tardivement. Cela est dû tout d'abord à une très longue période de conception, avec beaucoup de retours en arrière. C'est également dû en partie à notre exigence car cette dernière année à Gobelins représentait pour nous un moment à part.

En effet, le besoin de profiter des derniers instants ensemble s'est parfois heurté à la rigueur du projet. Le fait d'avoir une salle séparée a été à double tranchant : d'un côté, elle nous offrait un cadre propice au travail d'équipe mais de l'autre, elle est vite devenue un lieu de passage et de détente pour toute la promotion et parfois pour l'équipe pédagogique (cc Mathieu). De nombreux jeux parallèles ont émergé dans cette salle, tous ayant leurs propres règles. Cela a renforcé la cohésion, mais a aussi perturbé notre rythme de production.

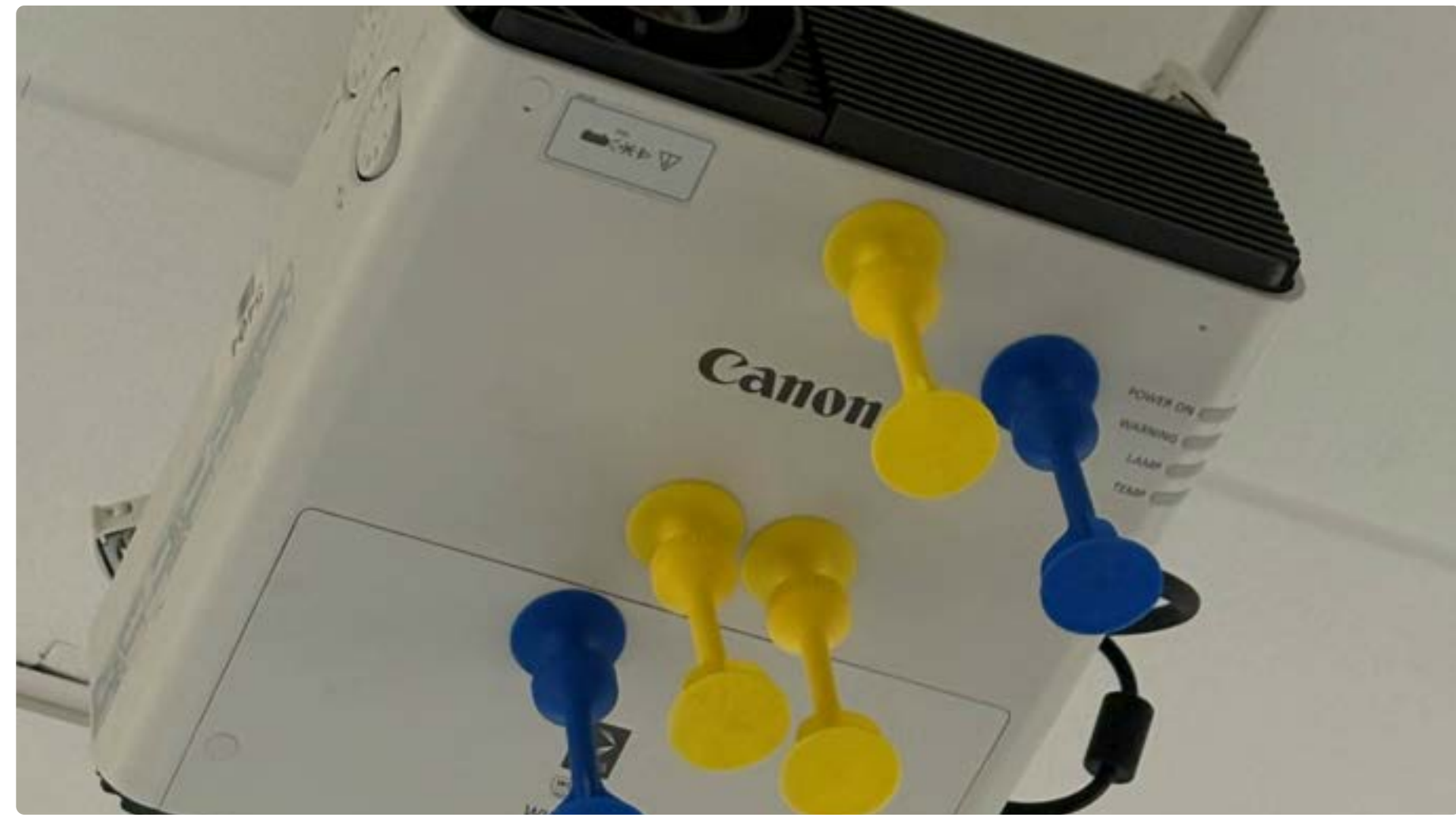
Mention spéciale à la "table à goûter", véritable lieu de ralliement quotidien, qui nous a offert autant de réconfort que de distractions (et de sucre).

Son & pédagogie

Impact & viabilité

Bilan

Notes de fin



REMERCIEMENTS

Remerciements spéciaux à Mathieu, Apolline, Fabienne, Marion, Xavier, Alex, Adrien, Karen, Gobelins.

phronesis™

merci :)