Descriptif des applications numériques

Balades

Mathématiques et informatique à l'air libre



Maison des Mathématiques et de l'Informatique - Octobre 2026

Sommaire

I. Cadre de la prestation scénographique..... 3
La Maison des Mathématiques et de l'Informatique..... 3
Le projet de l'exposition « Balades »..... 3
Comité scientifique et coordination..... 4
Objet de la prestation..... 4
II. Contexte de l'exposition..... 6
Fil rouge..... 6
Objectifs..... 7
Public visé..... 7
Expérience de visite proposée..... 8
III. Description des applications..... 8
Description générale de l'exposition..... 9
Application 1 – « Point de rendez-vous »..... 10
Informations générales..... 10
Matériel informatique..... 10
Contexte scientifique..... 10
Objectifs..... 11
Vue générale de l'application..... 12
Description page par page..... 13
Application 2 – « Embouteillages »..... 20
Informations générales..... 20
Matériel informatique..... 20
Contexte scientifique..... 20
Objectifs..... 24
Vue générale de l'application..... 25
Descriptif page par page..... 26
Application 3 – « Pique-nique »..... 32
Informations générales..... 32
Matériel informatique..... 32
Contexte scientifique..... 32
Objectifs..... 33
Vue générale de l'application..... 33
Descriptif page par page..... 36
Application 4 – « Les lucioles »..... 44
Informations générales..... 44
Matériel informatique..... 44

Contexte scientifique.....44
Objectifs..... 49
Vue générale de l’application..... 49
Descriptif page par page.....52

I. Cadre de la prestation scénographique

La Maison des Mathématiques et de l'Informatique

La Maison des Mathématiques et de l'Informatique (MMI) de Lyon est un lieu de diffusion de ces deux sciences créé en 2012. Elle accueille, sur 400 m², environ 250 classes (du CM1 à la Terminale) et 5000 visiteurs et visiteuses grand public par an. Ses activités sont nombreuses et variées : ateliers, visites d'exposition, contes mathématiques, stages de vacances, etc. Nous renvoyons au site https://mmi.universite-lyon.fr/ pour plus de détails.

Les expositions colorent les saisons de la MMI. Nous alternons entre sujets de société et sujets plus inattendus. Les dernières expositions étaient :

- *Sous la surface, les maths* (sur les jeux vidéos et l'infographie, 2019 - 2021).
- *Entrez dans le monde de l'IA* (sur l'intelligence artificielle, 2021 - 2023).
- *Dans ma cuisine – Les mathématiques et l'informatique se mettent à table* (sur les liens entre cuisine, mathématiques et informatique, 2023 - 2025). Il est possible de consulter le dossier de présentation de cette exposition en ligne. Le coût global de cette exposition (scénographie, matériel, applications numériques, ...) était de 98000 € TTC.
- *Ça résonne !* (sur le son, 2025 - 2027). Il est possible de consulter le dossier de présentation de cette exposition en ligne. Le coût global de cette exposition (scénographie, matériel, applications numériques, ...) était de 121000 € TTC.

L'exposition « Balades – Mathématiques et informatique à l'air libre » sera présentée à la MMI entre septembre 2027 et juin 2029. Celle-ci sera accompagnée de nombreuses activités, ateliers, manifestations autour du thème de l'extérieur et des promenades. Elle se situe dans la catégorie des sujets plus décalés et inattendus, et traitera une variété de sujets scientifiques différents.

Le projet de l'exposition « Balades »

L'exposition sera installée à la MMI en septembre 2027. Elle sera ouverte au public de septembre 2027 à juin 2029.

Comité scientifique et coordination

Les commissaires scientifiques de l'exposition sont :

- **Eric Duchêne**, professeur des universités en informatique à l'Université Lyon 1 et au laboratoire LIRIS, directeur adjoint de la MMI ;
- **Aline Parreau,** chargée de recherche au CNRS, rattachée à l'Université Lyon 1 et au laboratoire LIRIS.

La commissaire de l'exposition est **Camille Beaudou**, chargée de médiation et d'animation à la MMI.

Le synopsis d'exposition a été imaginé par les trois commissaires, le directeur et l'équipe de médiation de la MMI (**Olivier Druet, Charlotte Avellaneda** et **Nina Gasking**).

Le comité scientifique de l'exposition est composé de chercheurs et chercheuses en mathématiques et informatique :

- **Jean-Baptiste Aubin**, maître de conférences en statistiques à l'INSA-Lyon et membre du laboratoire ICJ ;
- **Jim Délitroz**, doctorant à la HES-SO Valais/Wallis et rattaché à l'EPFZ (Suisse) ;
- **Sébastien Designolle**, chercheur INRIA, membre du laboratoire LIP ;
- **Olivier Druet**, directeur de recherche au CNRS, membre du laboratoire ICJ, directeur de la MMI ;
- **Hélène Léman**, chargée de recherche à INRIA et membre du laboratoire UMPA ;
- **Olga Paris-Romaskevitch,** chargée de recherche au CNRS, membre du laboratoire ICJ ;
- **Paolo Pistone**, chaire professeur junior à l'Université Lyon 1 et membre du laboratoire LIP.

Objet de la prestation

Nous décrivons dans ce document l'objet exact de la prestation du lot « Applications numériques ». Celui-ci représente :

- 4 applications à développer, qui vont respectivement dans les stands *Ville 1*, *Ville 3*, *Montagne 2* et *Nuit 1* du pré-programme muséographique.
- Du matériel numérique :
    - 1 écran tactile de 24" ou 27" (pour l'application 1 « Point de rendez-vous ») ;
    - 1 écran non-tactile de 24'' ou 27'' (pour l'application 2 « Embouteillages ») ;
    - 2 ordinateurs (1 pour l'application 1 – « Point de rendez-vous » et 1 pour l'application 2 « Embouteillages ») ;

    - 1 manette de jeu vidéo (pour l'application 2 « Embouteillages ») ;
    - 3 tablettes de 13'' ou 14'' (2 pour l'application 3 « Pique-nique » et 1 pour l'application 4 « Les lucioles »).

Le travail devra se faire en étroite collaboration avec les scénographes de l'exposition et l'équipe de la MMI en charge de l'exposition. On pourra consulter le pré-programme muséographique pour se faire une idée de l'exposition.

Le prestataire sera en charge :

- du développement des applications ;
- du graphisme des applications (en adéquation avec le graphisme de l'exposition) ;
- de l'installation des applications dans l'exposition ;
- de l'achat du matériel informatique nécessaire à ces applications (écrans, tablettes, ordinateurs), en adéquation avec le mobilier fabriqué pour l'exposition ;
- de fournir le code des applications à la MMI.

Le prestataire n'est pas en charge :

- de l'écriture des textes à intégrer dans l'application ;
- de l'achat du vidéoprojecteur pour l'application « Les lucioles ».

Il est demandé aux prestataires d'avoir l'usage le plus limité possible des outils d'intelligence artificielle (LLM et IA générative). Notamment, **tous les programmes devront pouvoir être explicités par leurs développeurs et développeuses**. Il s'agit d'une exigence de rigueur scientifique : nous avons besoin de nous assurer de la justesse des algorithmes et du code utilisés. Par ailleurs, **aucun élément visuel final ne devra être généré par des outils d'intelligence artificielle** (même partiellement).

Pour toute question ou pour organiser une visite de la MMI, vous pouvez contacter Camille Beaudou (camille.beaudou@universite-lyon.fr).

II. Contexte de l'exposition

Fil rouge

L'exposition « Balades » invitera le public à poser un regard mathématique ou informatique sur le monde extérieur. Que peut-on y comprendre grâce aux mathématiques ou à l'informatique ? Quels phénomènes du monde extérieur peut-on expliquer grâce à ces disciplines ? Y a-t-il des parallèles entre certains éléments de l'extérieur et des questions mathématiques ou informatiques ?
L'exposition regroupera des thèmes scientifiques variés : statistiques, optimisation, géométrie, algorithmique, modélisation, analyse… Nous souhaitons susciter la curiosité du public sur les mathématiques, l'informatique, et/ou des éléments du monde extérieur. Un autre objectif sera de montrer que ces disciplines scientifiques existent hors-les-murs, en dehors des salles de classe ou des laboratoires.

L'exposition se découpera en quatre balades explorant des univers différents (la ville, la montagne, la mer…). Chaque balade, composée de plusieurs stands, proposera au public de poser un regard mathématique ou informatique sur des éléments liés à l'univers choisi. Nous souhaitons que chaque univers soit plaisant et invite à l'imagination (« je pars en balade dans tel lieu, je vais y découvrir tel élément »). Chaque balade aura une identité marquée le différenciant clairement des balades voisines, tout en gardant une cohérence garantissant l'unité de l'exposition.

Les balades sont les suivantes :

- **Balade en ville** – Un groupe d'amis et amies se donne rendez-vous en ville pour aller se promener ensemble, en passant par des lieux qu'ils et elles apprécient.
- **Balade en mer** – Un trésor est caché au large d'une mystérieuse île. La carte indique son emplacement, mais elle n'est pas très précise…
- **Balade en montagne** – Ce week-end, il fait beau. L'occasion de profiter de la montagne, pour y prendre des photos et partager un pique-nique.
- **Balade de nuit** – Quelles surprises nous réserve une balade dans le noir ? Nous partons à la découverte d'un groupe de lucioles au comportement presque magique. En chemin, des chercheurs et chercheuses nous parlent de mathématiques et d'informatique les yeux fermés.

Chaque balade regroupera deux ou trois stands, traitant d'un sujet scientifique distinct. A chaque stand, une activité de manipulation accompagnera la présentation du sujet et permettra de le découvrir et de mieux l'appréhender. Certaines manipulations seront entièrement physiques (pas d'écran), d'autres prendront la forme d'applications

numériques. Expérimenter est au cœur des principes de médiation de la MMI. En effet, la MMI aime à voir les mathématiques et l'informatique comme des sciences expérimentales où l'essai, l'erreur, le tâtonnement, sont fondamentaux pour avancer dans la compréhension, loin des images figées, froides et implacables de la rigueur et des démonstrations.

Objectifs

Les objectifs de l'exposition sont les suivants :

- Montrer les mathématiques et l'informatique autrement en donnant une grande place à la **manipulation**. Notamment, on souhaite présenter ces sciences comme des **disciplines vivantes et en mouvement**, pouvant aussi s'exercer à l'extérieur, et liées au monde qui nous entoure (sans pour autant être uniquement utilitaires).
- Montrer et faire faire des mathématiques et de l'informatique dans **un contexte plaisant, sollicitant l'imagination** et y prendre autant de plaisir que lorsque l'on (se) raconte une histoire ou que l'on part en promenade.
- De surprendre le public et d'éveiller la **curiosité scientifique** à partir d'éléments venant de l'extérieur.
- Montrer la **diversité** des mathématiques et de l'informatique.
- En dehors des mathématiques et de l'informatique : **faire voyager** le public dans des lieux qu'il ne côtoie pas forcément au quotidien (par exemple, la mer pour les personnes vivant en ville).

Public visé

Le public visé est à la fois un **public scolaire (de la cinquième à la Terminale) et un public familial (à partir de 10 ans)**. Le ton sera pensé pour des jeunes adolescents et adolescentes. Cependant, certains stands auront un contenu scientifique adressé à des élèves de fin de lycée : ils et elles devront donc également se sentir à leur place dans l'exposition.

Il convient de noter qu'à la MMI, trois à quatre fois plus de scolaires que de grand public viennent visiter l'exposition.

Expérience de visite proposée

L'exposition est conçue pour être visitée **avec un médiateur ou une médiatrice**. Les différents espaces (balades) doivent être compréhensibles et utilisables en autonomie mais il y aura toujours une médiation humaine. Le médiateur ou la médiatrice pourra aider pour les manipulations, apporter plus d'explications sur les stands, répondre aux questions du public, etc.

Dans l'exposition, le public est **mis en activité** par l'intermédiaire de manipulations (physiques ou numériques). La plupart des manipulations sont accompagnées de panneaux explicatifs, et d'une consigne.

Deux types de visites sont proposés :

- Pour les scolaires, beaucoup de visites ont lieu sur un temps d'1 h 15 en demi-classe pendant que l'autre demi-classe participe à un atelier ;
- Les visites guidées standard dirent environ 1 h 30 et peuvent accueillir soit du grand public, soit une classe entière (rarement).

Une visite type sera organisée ainsi :

- Une présentation générale (en groupe) de l'exposition, et un focus sur l'un des stands avec des explications scientifiques plus poussées (le stand sera choisi en fonction du public présent), pendant environ 30 minutes ;
- une visite libre de 45 minutes environ avec présence d'un médiateur ou d'une médiatrice.

Un créneau de visite de l'exposition accueille généralement entre 15 et 20 personnes. Dans des cas plus rares, ce nombre peut monter jusqu'à une trentaine de personnes.

III. Description des applications

Description générale de l'exposition

L'exposition est composée de 5 espaces :

- **Une zone « Point de départ ».** Elle n'a pas vocation à être un espace de manipulation, mais plutôt à aider à la cohérence scénographique de l'exposition. Il s'agit donc uniquement de mobilier.
- **Une « balade en ville »**, constituée de 3 stands (dont deux manipulations numériques).
- **Une « balade en mer »**, constituée de 3 stands.
- **Une « balade en montagne »**, constituée de 3 stands (dont une manipulation numérique).
- **Une « balade de nuit »**, constituée de 2 stands (dont une manipulation numérique).

Ceci n'est pas un cahier des charges contractuel. Il s'agit d'une description des scénarios envisagés afin de se faire une idée du travail en termes de développement. De légères modifications pourront être apportées en fonction de la scénographie, du matériel (en particulier la taille des écrans), de l'ambiance générale de l'exposition. Les scénarios définitifs et donc les cahiers des charges seront écrits par la MMI après travail avec les équipes chargées de la scénographie et du développement des applications.

Application 1 – « Point de rendez-vous »

L'histoire : *Mes proches et moi habitons dans différents quartiers de la ville. Nous cherchons le meilleur endroit pour nous retrouver.*

Informations générales

Cette application sera intégrée à la « balade en ville », dans le stand Ville 1 du pré-programme muséographique. Outre l'application, le stand comprendra un panneau explicatif, ainsi qu'une manipulation physique courte faisant office d'introduction.

Matériel informatique

★ Un écran tactile (24" ou 27") ;
★ Un ordinateur.

Contexte scientifique

Dans ce stand, on propose de réfléchir à la définition de centre pour un ensemble de points. Comme assez souvent en mathématiques, il n'y a pas une seule définition possible mais plusieurs en fonction des critères que l'on considère et de notre objectif. Dans le cas du centre, beaucoup de définitions sont possibles. Certains ont par exemple plus d'utilité en statistiques, d'autres sont plus faciles à calculer et sont utilisés pour cela.

Dans la manipulation, on se place dans la situation suivante : un groupe de personnes, partant de différents endroits, cherche un point « central » pour se retrouver. On va se concentrer sur trois définitions du centre :

- **Le centre minimax** : c'est le centre du plus petit cercle contenant tous les points. Autrement dit, c'est l'endroit qui minimise la plus grande distance séparant le centre du point le plus éloigné. Ici, peu importe qu'il y ait beaucoup de points à un endroit : ce qui compte, c'est de ne pas avoir de points trop éloignés du centre. L'inconvénient de ce centre est qu'il est très influencé par un point éloigné, même si tous les autres sont regroupés au même endroit.
  Pour calculer ce centre : on peut d'abord remarquer que le plus petit cercle contenant tous les points passe forcément soit par au moins trois points, soit par deux points qui sont diamétralement opposés. Dans le premier cas, c'est le centre du *cercle circonscrit* de ces trois points. Ses coordonnées peuvent se calculer à l'aide de formules assez simples. Dans le deuxième cas, le centre est le milieu des deux points.

Pour trouver le centre, on peut alors prendre chaque triplet de points et chaque paire de points, calculer dans chaque cas le centre du cercle qui passe par ces points et garder celui qui minimise le rayon. On peut réduire la recherche en ne prenant que les points les "plus à l'extérieur".

- **Le centre médian** : c'est celui qui minimise la somme des distances entre le centre et les points autour. Il a tendance à se rapprocher des gros groupes de points.
  Pour calculer ce centre : ce n'est pas évident malgré sa simplicité dans la définition. Il n'y a pas de méthode exacte. La méthode la plus utilisée est l'algorithme de Weiszfeld qui trouve une bonne approximation du point par itérations successives. Cet algorithme pourra être fourni et explicité par la MMI si besoin.

- **Le centre de gravité** : formellement, il minimise la somme des distances au carré. Cela signifie qu'un point éloigné aura plus de poids qu'un point proche, mais que beaucoup de points au même endroit rapprochent le centre. Ce centre n'est pas trop influencé par un seul point s'il y a beaucoup de points.
  Pour calculer ce centre : il s'agit du point avec comme coordonnées la moyenne des abscisses et des ordonnées.

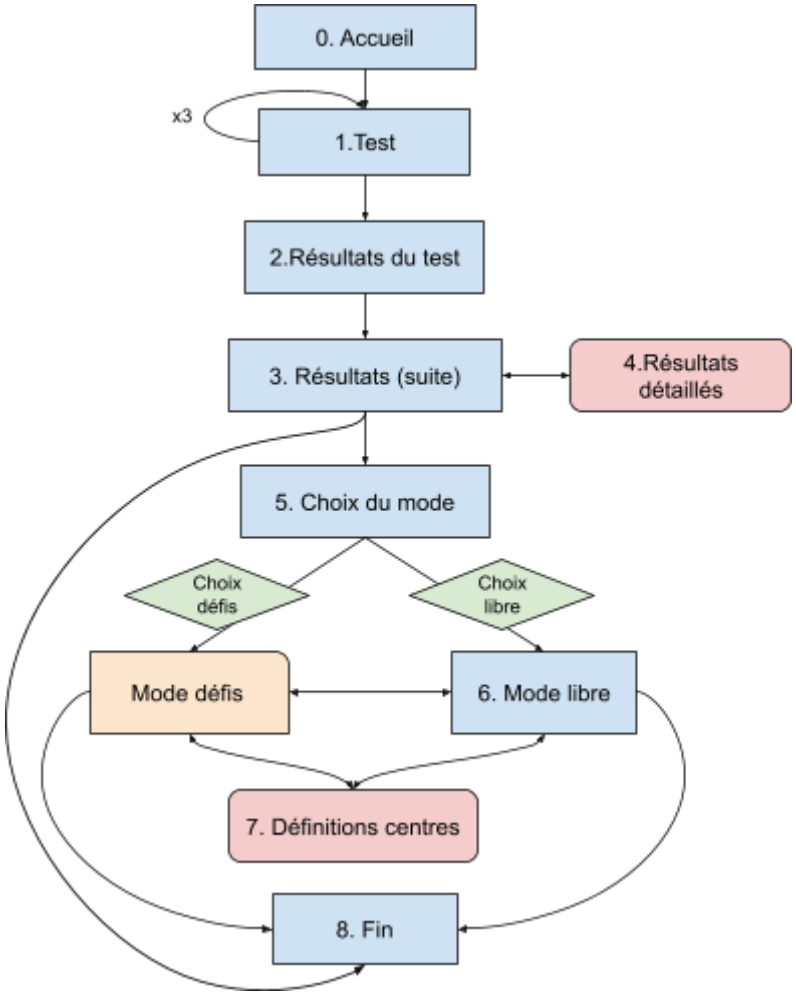
Les algorithmes permettant de calculer la position des centres pourront être donnés par la MMI si nécessaire.

Objectifs

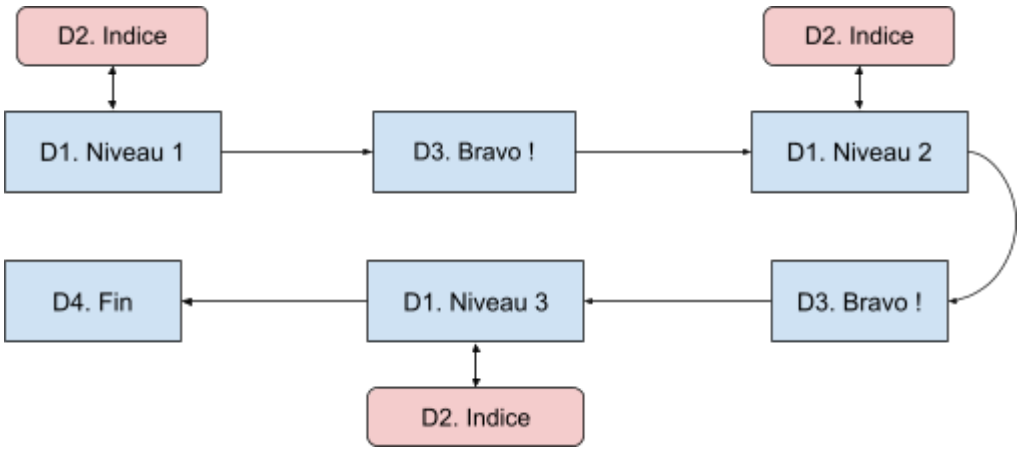
★ Découvrir qu'en mathématiques, il peut y avoir plusieurs définitions différentes valides pour une même notion. Souvent, on choisit la définition selon le problème que l'on regarde et notre objectif.
★ Découvrir quelques définitions possibles pour la notion de centre, leurs intérêts et caractéristiques.

Vue générale de l'application

Scénario général



Mode défis



Précisions pour toute l'application :

- Tout le discours scientifique de l'application tourne autour de 3 définitions possibles du centre d'un nuage de points. Dans la description de l'application et les illustrations, ils apparaissent avec les couleurs suivantes : rouge (centre minimax), vert (centre de gravité) et bleu (centre médian). Les couleurs ne sont pas définitives et seront à décider en accord avec la scénographie.
- Les textes ne sont pas définitifs.
- A tout moment, présence d'un bouton « Retour au début » qui renvoie à l'accueil.
- A tout moment, au bout d'un certain temps d'inactivité, on revient automatiquement à l'accueil.
- Toutes les pages comportent un bouton « Médiation » qui permet, après avoir saisi un code secret, d'accéder directement au mode libre (page 6 du scénario général).
- Dans toute l'application, dès qu'une illustration avec les points est utilisée, elle devra être en accord avec l'illustration du panneau (couleurs, symboles utilisés).
- Il faudra prévoir une façon de faire un reset des résultats du quiz.

A charge de la MMI :

- Les positions de tous les points pour le quiz, des centres « objectif » dans le mode défi, des points à figer pour les indices du mode défi ;
- Les textes ;
- Le contenu scientifique concernant le calcul de la position des différents centres.

Description page par page

Scénario général

0. Page d'accueil

Une page d'accueil de l'application, simple, avec un peu de texte, un visuel et avec un bouton.

1. Test

Le public va voir cette page 3 fois. En effet, celle-ci propose successivement 3 nuages de points où le public doit estimer à chaque fois où est le centre. Cette partie est un peu pensée à la façon d'un test de personnalité proposé par certains magazines, un « Quel centre es-tu ? » en quelque sorte. Le ton (notamment des résultats de ce test) est donc volontairement décalé. On pourra donc jouer sur cette idée lors du design de l'application, si cela paraît pertinent.

Éléments présents sur la page :

- Un titre ;
- Une image avec un ensemble de points, tous de la même couleur ;

- Du texte : « Voilà où habitent tes proches. D'après toi, où est le meilleur endroit pour se retrouver ? » ;
- Un bouton « Valider » ;
- Un indicateur permettant de situer le niveau de progression (« Question 1 sur 3 » par exemple).

Interaction public : La personne clique sur l'endroit qui lui semble le mieux. Une fois qu'un point est placé, le bouton « Valider » apparaît (ou devient cliquable). La personne peut déplacer son point si elle le souhaite, puis, quand elle est satisfaite, elle clique sur « Valider ». Lorsque la personne clique sur valider, soit un autre test apparaît (3 itérations au total), soit on passe à la page 2.

Remarques :
- Le choix du centre est libre.
- Il faut retenir le choix fait par la personne ainsi que les choix faits par les personnes précédentes.
- Il faut calculer et retenir la distance à chaque centre.

2. Résultats du test

On indique à la personne de quel centre (centre minimax, centre médian, centre de gravité) elle est le plus proche. Le résultat est obtenu en calculant pour chaque centre la somme des distances obtenues à chaque étape. Le centre qui a le plus petit score gagne. En cas d'égalité (peu probable avec suffisamment de précision), on tirera au sort le centre qui gagne.

Éléments présents sur la page :
- Un titre ;
- Du texte indiquant le résultat de la personne. Par exemple : « **Tu es plutôt *centre de gravité***. Pour toi, ce qui est important, c'est de veiller au bon équilibre du groupe. Il faut prendre en compte les personnes qui habitent loin, tout en évitant trop de déplacements. Un exercice d'équilibriste ! » ;
- Un bouton « Suite ».

Interaction public : passage à la suite grâce au bouton.

3. Résultats (suite)

Cette page permet de présenter les autres résultats qui étaient possibles pour le test (c'est-à-dire les deux autres centres).

Éléments présents sur la page :
- Un titre ;
- Deux zones de texte, comportant chacun une description des deux autres centres possibles (en reprenant l'exemple de la page 2, on aurait ici le **centre minimax** et le **centre médian**) ;
- Un bouton « Résultats détaillés » ;

- Un bouton « Terminer » ;
- Un bouton « Jeux bonus ».

Interaction public :
- Le bouton « Résultats détaillés » ouvre une pop-up (page 4).
- Le bouton « Jeux bonus » envoie à la suite (page 5).
- Le bouton « Terminer » permet de finir la manipulation, en passant à la page 8.

4. Résultats détaillés

Éléments présents sur la page :
- Un titre ;
- Un carrousel de 3 images, reprenant les 3 nuages de points utilisés lors du test (page 1). Sur chaque image, on voit :
    - Le nuage de points ;
    - L'endroit où la personne a voté ;
    - Le centre minimax, le centre de gravité, et le centre médian (dans la bonne couleur) ;
    - Une *heatmap* représentant les endroits où les gens ont voté.
- Une légende, avec les informations suivantes :
    - Couleurs des points (les 3 centres, le vote de la personne, et le nuage de points) ;
    - Couleurs de la *heatmap*.
- Un bouton pour fermer la pop-up.

Interaction public :
- On peut faire défiler les 3 images du carrousel.
- Un bouton permet de fermer la pop-up, et de revenir à la page 3.

Remarque : la granularité et le mode de calcul des heatmaps sera fourni par la MMI.

5. Choix du mode

Une page de menu, avec un petit texte de transition. Cette page permet de choisir entre le mode « défi » et le mode « libre ».

6. Mode libre

Cette page permet de placer, déplacer et supprimer librement des points. Les 3 centres s'affichent et s'actualisent automatiquement en fonction des points placés. Pour se faire une idée, on peut voir un prototype ici : https://gbagan.github.io/center/ (choisir « mode test »).

Éléments présents sur la page :
- Un titre ;
- Une consigne (une ou deux lignes) ;
- Un cadre où la personne pourra placer ses points ;
- Une légende (indiquant la couleur des centres et leurs noms) ;
- Un bouton « Réinitialiser » ;

- Un bouton « Aller aux défis » ;
- Un système permettant de choisir quels centres sont affichés (cases à cocher par exemple).

Interaction public :
- La personne peut ajouter de nouveaux points dans le cadre. Elle peut aussi déplacer et supprimer des points déjà créés. À chaque action de la personne, les centres sont recalculés et leur emplacement est modifié dynamiquement.
- Lorsque l'on touche le point représentant le centre minimax, le cercle englobant tous les points s'affichent s'affiche.
- Lorsque l'on touche le centre médian, des traits s'affichent pour relier ce point à tous les autres.
- La personne peut choisir quels centres s'affichent.
- Le bouton « Réinitialiser » permet de supprimer tous les points.
- Le bouton « Aller aux défis » envoie vers le mode défis.

Remarque : il peut arriver que les centres coïncident entre eux ou avec des points. Dans ce cas, il faut que l'affichage soit clair.

7. Définition des centres

Une page pop-up permettant d'afficher les définitions des centres.

Éléments présents sur la page :
- Un titre ;
- 3 zones de texte (une pour chaque centre), avec chacune un titre, quelques ligne de texte, et éventuellement une illustration ;
- Un bouton pour fermer la pop-up.

Interaction public : fermeture de la pop-up.

8. Fin

Une page très simple, pour marquer la fin de l'activité. Elle sera composée d'un titre et d'une ou deux lignes de texte, ainsi que d'un bouton pour revenir à la page d'accueil.

Mode défis

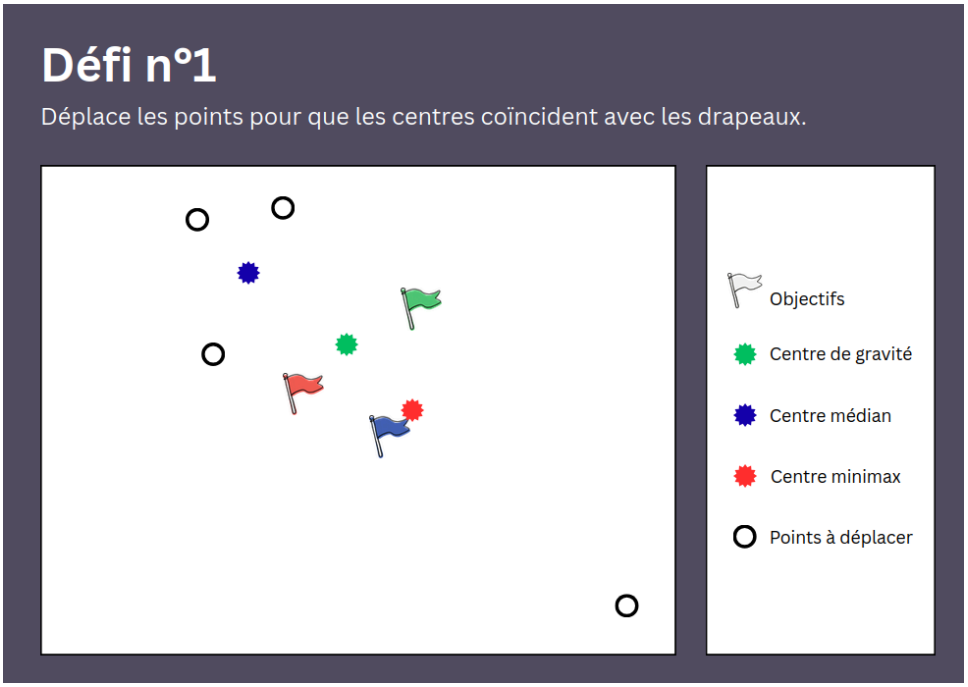
Ce mode est constitué de 3 défis successifs, avec un niveau croissant de difficulté. Le jeu, similaire à chaque niveau, est le suivant : 3 centres « objectif » sont fixés dans un cadre. Ces centres sont représentés par des drapeaux dans la suite, mais cela peut être un autre symbole. Comme leur nom l'indique, ce sont les objectifs à atteindre. Un certain nombre de points (4 ou 5) sont déjà placés dans le cadre. Le but du jeu est de déplacer ces points pour que les 3 centres obtenus (calculés dynamiquement) coïncident avec les centres « objectif ».

D1. Niveau 1/2/3

Éléments présents sur la page :
- Un titre ;

- Une courte consigne : « Déplace les points pour que les centres coïncident avec les drapeaux » ;
- Un cadre où la personne pourra jouer et où sont déjà présents trois drapeaux de couleur (qui sont les objectifs à atteindre et qui restent fixes), des points (qui peuvent être déplacés) et leurs centres actuels (qui sont calculés automatiquement) ;
- Une légende (indiquant la couleur des centres et leurs noms). Pour chaque centre, un petit bouton « Plus d'informations » permet d'afficher la définition ;
- Un système permettant de choisir quels centres sont affichés (cases à cocher par exemple) ;
- Des boutons :
    - « Aller au mode libre » ;
    - « Indice 1 » (qui devient cliquable au bout d'un certain temps de recherche) ;
    - « Indice 2 » (qui devient cliquable une fois le 1er indice utilisé) ;
    - « Solution » (qui devient cliquable une fois le 2ème indice utilisé).

Pour clarifier une partie de la liste ci-dessus, voilà un exemple de ce à quoi pourrait ressembler cette page. *Cette illustration n'est qu'indicative, elle n'est en rien une demande en termes de design final de l'application.*



Interaction public :

- Un clic sur « Aller au mode libre » envoie à la page 6 du scénario général.
- Un clic sur les boutons « Indice 1 » ouvre la page D2 (une pop-up).
- Un clic sur le bouton « Indice 2 » fixe certains points, qui ne sont alors plus déplaçables par la personne.

- Un clic sur le bouton « Solution » fixe tous les points dans une solution possible. Dans ce cas, cela fait apparaître 3 boutons : « Défi suivant », « Mode libre », « Terminer le jeu ».
- La personne peut déplacer les points. Cela modifie l'emplacement des 3 centres. Comme dans le mode libre, cela peut générer des superpositions qu'il conviendra de gérer pour conserver la lisibilité.
- La personne peut choisir quels centres s'affichent (avec des cases à cocher par exemple).
- Le bouton « Plus d'informations » de chaque centre permet d'afficher une petite bulle informative avec une courte définition du centre concerné.

Lorsque les centres « à placer » se retrouvent tous suffisamment proches des centres « objectif », la pop-up « Bravo ! » s'ouvre automatiquement.

D2. Indice

Éléments présents sur la page :

- Un titre ;
- Quelques lignes de texte. Par exemple : « Essaie d'abord de fixer le centre de couleur rouge avec 2 points. Ensuite, ne bouge plus ces points et fixe les deux centres restants. » ;
- Un bouton pour fermer la fenêtre.

Interaction public : fermeture de la pop-up.

D3. Bravo !

Cette pop-up s'ouvre automatiquement une fois la solution à un défi trouvée.

Éléments présents sur la page :

- Un titre ;
- Eventuellement, une très courte ligne de texte ;
- Des boutons ;
    - « Défi suivant » ;
    - « Mode libre » ;
    - « Finir le jeu ».

Interaction public :

- Le bouton « Défi suivant » emmène vers le défi 2 ou 3.
- Le bouton « Mode libre » emmène à la page 6 de l'organigramme général.
- Le bouton « Finir le jeu » emmène à la page 8 de l'organigramme général.

D4. Fin

Cette pop-up s'ouvre automatiquement une fois la solution au niveau 3 trouvée.

Éléments présents sur la page :

- Un titre ;

- Eventuellement, une très courte ligne de texte ;
- Un bouton « Mode libre » ;
- Un bouton « Finir le jeu ».

Interaction public

- Le bouton « Mode libre » emmène à la page 6 de l'organigramme général.
- Le bouton « Finir le jeu » emmène à la page 8 de l'organigramme général.

Application 2 – « Embouteillages »

L'histoire : *Qui dit grande ville dit embouteillages ! Bloquée dans les bouchons, je me demande si cela aurait pu être évité en organisant les routes autrement.*

Informations générales

Cette application sera intégrée à la « balade en ville », au stand Ville 3 du pré-programme muséographique. Le stand comprendra un panneau explicatif pour compléter les informations fournies dans l'application.

Matériel informatique

- ★ Un écran (24" ou 27") non-tactile ;
- ★ Un ordinateur ;
- ★ Une manette de jeu vidéo.

Contexte scientifique

Dans de nombreux contextes, des choix individuels rationnels (d'optimisation individuelle) ne mènent pas à une optimisation collective. Une des situations les plus connues est le dilemme du prisonnier : deux suspects sont interrogés et ils peuvent soit dénoncer l'autre, soit se taire. S'ils se taisent tous les deux, ils sont condamnés à 2 ans de prison. Si l'un dénonce l'autre alors que l'autre se tait, celui qui a dénoncé ne va pas en prison, l'autre est condamné à 10 ans. S'ils se dénoncent mutuellement, ils vont tous les deux 4 ans en prison. Quelle que soit la façon dont je retourne le problème, j'ai intérêt à dénoncer. Car quoi que fasse l'autre, ce sera meilleur pour moi :

- s'il me dénonce aussi, je fais 4 ans alors que j'en aurais fait 10 si je ne l'avais pas dénoncé ;
- s'il ne me dénonce pas, je ne vais pas en prison alors que j'aurais fait 2 ans si je ne l'avais pas dénoncé.

Et, pourtant, globalement, les deux suspects feraient mieux de se mettre d'accord et de ne pas se dénoncer puisqu'ils feraient deux ans de prison, au lieu des quatre lorsqu'ils se dénoncent mutuellement. Même si cette situation n'est pas très réaliste, elle est le modèle type du choix individuel rationnel entraînant un choix collectif catastrophique.

Dans cette manipulation, on va voir une autre situation, plus réaliste, où ce même phénomène se produit. Imaginons que, sur un réseau routier, chaque automobiliste suive une application d'optimisation du temps de transport (comme Waze, par exemple). À chaque carrefour, l'application indique la route la plus courte pour arriver à destination, et la personne suit la recommandation. C'est ce que nous faisons toutes et tous !

Or, le paradoxe de Braess donne des situations où, si on considère des trajets allant de A à B, construire une route qui rend le trajet individuel plus court peut mener à une perte de temps collective. Autrement dit, le temps de trajet moyen devient plus grand que si la route était absente. Chaque automobiliste cherche à optimiser son temps de trajet mais, globalement, en utilisant cette route, on la sature et on perd du temps. L'algorithme précis de la manipulation est décrit ci-dessous.

Comportement des routes et des voitures

Le comportement des divers éléments est donné pour comprendre l'algorithme à écrire. Il n'est pas nécessaire de tout comprendre parfaitement à ce stade : tout pourra ensuite être discuté et explicité avec la MMI. Ces informations ont pour but de clarifier l'algorithme demandé, et d'aider à évaluer la demande.
Les paramètres exacts seront fournis par la MMI, éventuellement après quelques tests.

Le réseau routier

Il s'agit d'un graphe orienté représentant les routes. Chaque route contient en information le nombre de voitures présentes sur celle-ci. Cette information est actualisée à chaque pas de temps. La couleur de cette route dépendra du nombre de voitures sur la route (ou du temps de parcours, voir plus bas) : vert s'il n'y a pas beaucoup de véhicules, puis jaune, orange, et enfin rouge quand il y a beaucoup de voitures. Les seuils de couleur seront à déterminer.

Les routes

Il y a deux types de routes :

- Les grosses routes (sans saturation). Celles-ci ont un temps de parcours fixe, qui ne dépend pas du nombre de voitures qui l'empruntent. Elles sont toujours représentées en vert, la circulation y est toujours fluide.
- Les petites routes (avec saturation). Le temps de parcours dépend du nombre de voitures (N) qui sont dessus. Le comportement sera le suivant : si N est inférieur à un seuil N_0, le temps est fixe T_0. Si $N > N_0$, le temps de parcours devient $T = T_0 + a \times (N - N_0)^2$. Il faut donc actualiser le temps de parcours à chaque pas de temps en fonction du nombre de voitures présentes sur la route. Les paramètres pour chaque route (T_0, a, N_0) seront précisés ultérieurement par la MMI.

Les voitures

Elles partent du sommet D à raison d'une par pas de temps (1 ms par exemple). Dans toutes les versions où il y a plusieurs voitures, il faudra en faire partir 500, 1000 ou plus.

Chaque voiture doit stocker : la route sur laquelle elle est, et son heure d'arrivée à la fin de la route (c'est-à-dire au carrefour suivant).

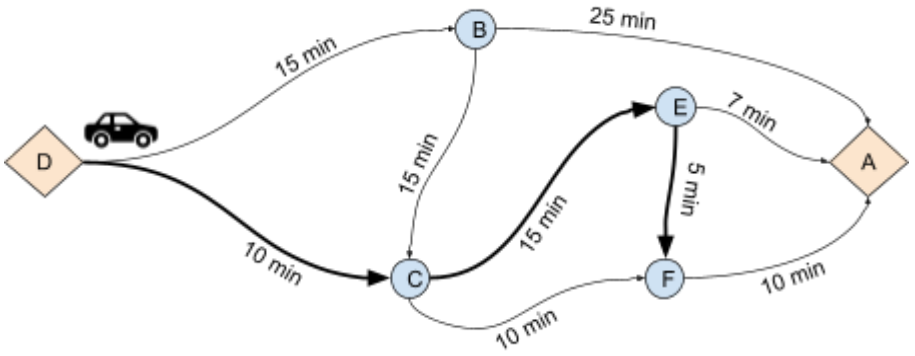
A chaque carrefour, la voiture choisit de prendre le plus court chemin (à l'instant où elle arrive au carrefour) : parmi tous les chemins possibles pour arriver à A depuis son carrefour, elle prend le plus rapide. On réactualise donc la route sur laquelle elle se trouve et son heure d'arrivée à la fin de cette route. Même si le temps de parcours de la route évolue une fois qu'elle est dessus, on ne change pas son temps d'arrivée : pas de prise en compte de l'évolution du trafic une fois que le choix a été fait. Par contre, au carrefour suivant, il est possible que le plus court chemin ait changé.

Quand la voiture est arrivée en A, on stocke son temps total de parcours. Pour cela, il faudra que l'heure de départ de chaque voiture soit stockée quelque part dans le programme.

Quand toutes les voitures sont arrivées en A, le score est le temps moyen de parcours de toutes les voitures.

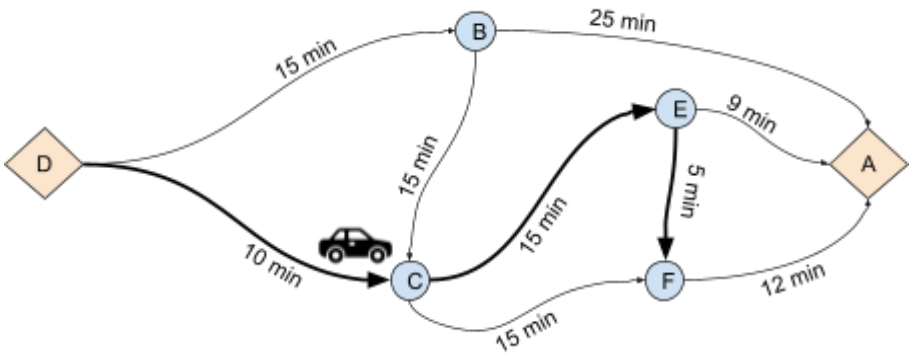
Voici un exemple sur un graphe simple pour illustrer le comportement de chaque voiture.

→ **Au temps T_0**, la voiture part de D :



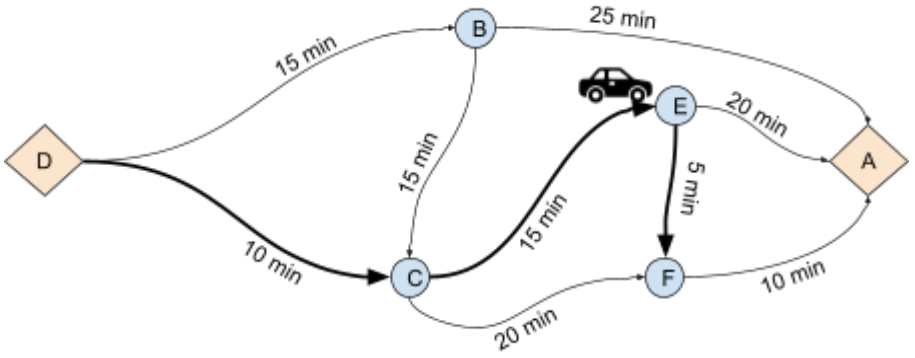
Comme le plus court chemin est donné, à ce moment-là, par D-C-F-A, on stocke la route [DC] et le temps d'arrivée au prochain carrefour $T_1 = T_0 + 10\text{min}$. On ajoute une voiture sur la route [DC].
Pour la voiture en question, il ne se passe rien pendant son temps de parcours entre T_0 et T_1 (600 secondes).

→ **Au temps T_1**, la voiture arrive en C :



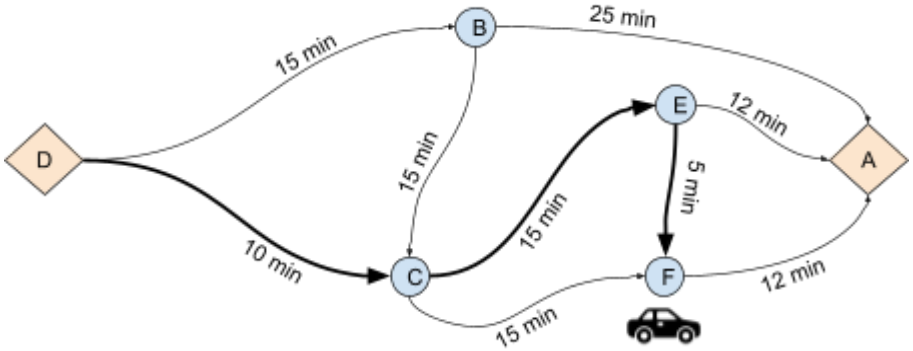
Les temps de parcours des petites routes ont changé car le nombre de voitures sur celles-ci a augmenté. Pour choisir, la prochaine route, on cherche le chemin le plus court pour aller à A, qui est, au temps T_1, C-E-A. On actualise donc la route de la voiture, qui est maintenant [CE] et le temps d'arrivée $T_2 = T_1 + 15$ min. On enlève cette voiture à la route [DC] et on la ajoute une à [CE].
Pour la voiture en question, rien ne se passe pendant 900 secondes, même si les routes changent de temps de parcours.

→ **Au temps T_2**, la voiture arrive en E :



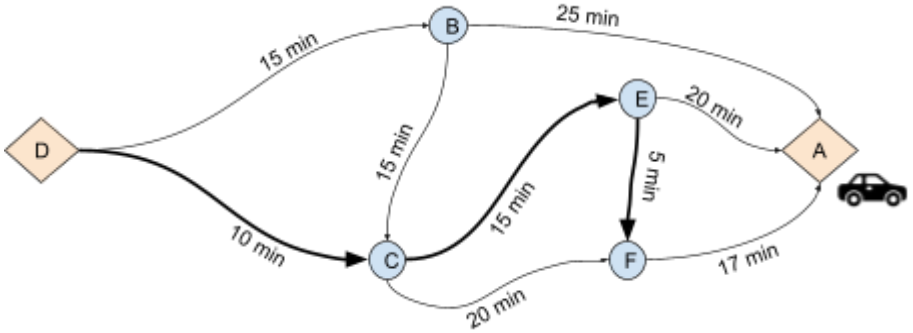
Le chemin le plus court pour aller à A est maintenant E-F-A car la route [EA] a saturé entre T_1 et T_2. On actualise donc la route de la voiture, qui est maintenant [EF] et le temps d'arrivée $T_3 = T_2 + 5$ min. On enlève également une voiture à la route [CE] et on en ajoute une à [EF].
Pour la voiture en question, rien ne se passe pendant 300 secondes, même si les routes changent de temps de parcours.

→ **Au temps T_3**, la voiture arrive en F :



Elle n'a plus le choix et va prendre la route [FA]. On actualise la route et le temps d'arrivée $T_4 = T_3 + 12$ min. On actualise aussi le nombre de voitures sur les routes [EF] et [FA].

→ **Au temps T_4**, la voiture arrive en A :



Même si le temps de parcours de la route [FA] a changé pendant que la voiture était dessus, on n'a pas touché au temps T_4. Cette actualisation ne concerne que les voitures arrivées en F après la voiture qu'on considère. On ajoute le temps total de parcours de la voiture, $T_4 - T_0$, au temps total de parcours de toutes les voitures.

On arrête lorsque toutes les voitures sont arrivées.

Pour résumer, l'algorithme de la simulation consistera en une boucle s'arrêtant une fois toutes les voitures arrivées à destination. Une itération de cette boucle équivaudra à un pas de temps, c'est-à-dire à l'actualisation de l'état des routes et aux choix des voitures arrivées sur un sommet à ce pas de temps.

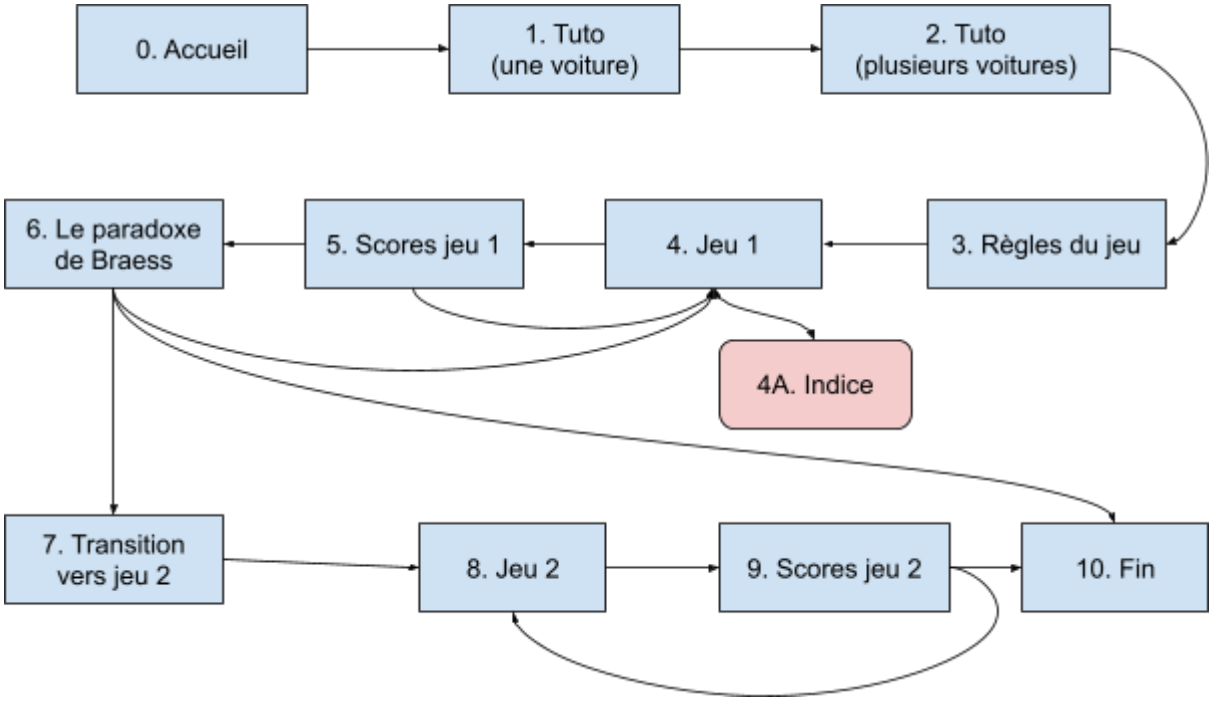
Objectifs

★ Découvrir un exemple de modélisation mathématique d'une situation (le flux de circulation).

★ Mettre en lumière que des choix individuels optimaux ne mènent pas à un choix collectif optimal et montrer qu'une régulation peut améliorer le choix collectif.

Vue générale de l'application

Scénario



Précisions pour toute l'application

- L'interaction dans cette manipulation se fait grâce à une manette de jeu vidéo, y compris les passages de page en page. Il faut donc prévoir des petits encarts sur les pages expliquant le rôle des divers boutons de la manette.
- Les textes et schémas ne sont pas définitifs.
- À tout moment, présence d'un bouton « Retour au début » qui renvoie à l'accueil.
- À tout moment, au bout d'un certain temps d'inactivité, on revient automatiquement à l'accueil.
- Un tableau des scores initial doit être présent. S'y ajoutent les scores des visiteurs et visiteuses de l'exposition. Il faudra prévoir la possibilité (pour le personnel de la MMI) de rétablir ponctuellement le tableau des scores initial. Cela pourra être fait en branchant un clavier et/ou une souris si nécessaire.

A charge de la MMI :

- Les cartes des réseaux routiers ;
- Les textes ;

- Le contenu scientifique concernant les simulations ;
- L'algorithme précis suivi par les voitures.

Descriptif page par page

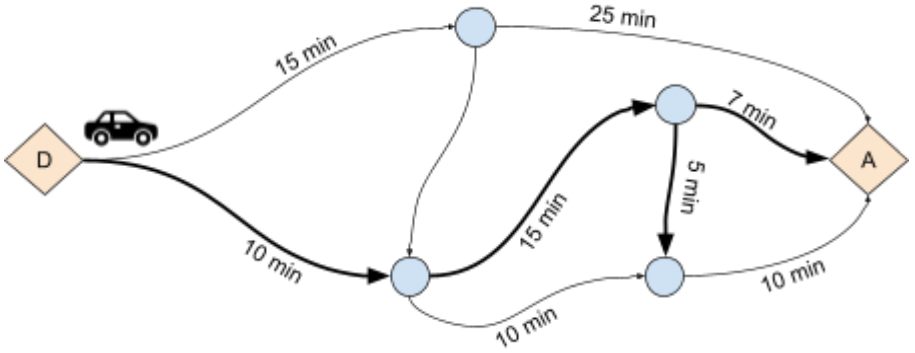
0. Accueil

Une page d'accueil de l'application, simple, avec un peu de texte, un visuel et avec un bouton permettant de lancer l'application.

1. Tuto (une voiture)

Éléments présents sur la page :

- Un titre ;
- Un très court texte de contexte / consigne ;
- Un cadre avec :
    - Une route découpée en plusieurs tronçons et plusieurs étapes. On peut distinguer 2 types de routes : des « grosses routes » (qui ne saturent jamais, il n'y a pas de capacité maximum) et des « petites routes » (dont le temps de parcours change quand on atteint un certain nombre de voitures). L'exemple ci-dessous permet de se faire une idée de ce à quoi cela pourrait ressembler.
    - Une voiture qui va se déplacer le long des routes, selon les choix de l'utilisateur ou utilisatrice.
    - Une bulle d'information pour la voiture qui va s'ouvrir automatiquement.



Éléments dynamiques :

On imagine cette page et la suivante un peu comme les « tutoriels » de certains logiciels ou jeux vidéos : des petites bulles d'informations sur les éléments importants s'ouvrent successivement. Chaque bulle contiendra un titre (le nom de l'élément), deux ou trois informations.

Pour cette première page, on a une bulle uniquement pour la voiture, avec un texte semblable à celui-ci : « La conductrice veut aller du point D au point A, le plus vite possible. À chaque croisement, elle décide quelle route prendre pour arriver rapidement. »

Une action sur la manette permet de fermer la bulle et de lancer le jeu.

Une fois le jeu lancé, la voiture avance le long des routes choisies. À chaque intersection, la personne choisit, à l'aide de la manette, le chemin qu'elle souhaite emprunter. Le problème est volontairement très facile, car le but n'est pas d'y passer du temps.

Interaction public :

- Fermeture de la bulle d'information ;
- Choix de routes à emprunter à chaque intersection ;
- Une fois la voiture arrivée à destination :
    - Si la route choisie était la bonne : un message apparaît (par exemple « Bravo, tu as choisi le meilleur chemin ! ») et permet de passer à la page suivante.
    - Sinon : un message apparaît (par exemple : « Était-ce vraiment le chemin le plus court ? ») et demande à la personne de refaire le jeu. Dans ce cas, le plan reste le même.

2. Tuto (plusieurs voitures)

Éléments présents sur la page :

- Un titre ;
- Quelques lignes de texte, expliquant qu'on passe maintenant à une circulation avec beaucoup plus de véhicules. Chaque individu (c'est-à-dire chaque voiture) cherche à relier les points D et A le plus rapidement possible ;
- Un cadre avec le même plan qu'en page 1. Sur ce cadre s'ouvrent successivement les deux bulles d'informations suivantes décrites ci-après. Puis, on lance une animation montrant ce qu'il se passe sur le plan avec un grand nombre de voitures.
    - Pour une « grosse route » (en gras sur le schéma) : « Cette route est une voie rapide, elle ne sature jamais. Temps de parcours : 5 minutes. » ;
    - Pour une « petite route » : « Cette route ne comporte qu'une seule voie. Temps de parcours : 8 minutes. Ce temps de parcours augmente très vite au-delà de 20 véhicules présents sur la route. ».
- Un cadre avec le temps moyen de parcours.

Description de l'animation :

La personne peut passer de la bulle d'information « grosse route » à la suivante (« petite route »). Une fois ces deux contenus lus, on lance automatiquement une animation dans le cadre où se situe le plan.

Sur ce plan, on « lance » un certain nombre de voitures (1000 voitures lancées les unes après les autres, par exemple 1 toute les 10 ms). L'algorithme à suivre est décrit plus haut. L'animation sera toujours la même, elle peut donc être pré-enregistrée et n'a pas besoin d'être recalculée à chaque fois.

L'animation ne montre pas les voitures individuellement, mais l'état de saturation de chaque tronçon de route (par exemple, avec un code couleur, qui dépend du nombre de voitures), le temps de parcours de chaque tronçon (qui s'actualise au fur et à mesure).

Le temps moyen de parcours s'affiche au fur et à mesure, à partir du moment où la première voiture est arrivée à A. On souhaite également qu'une indication du volume de voitures soit indiqué, ainsi que la progression du nombre de voitures déjà arrivées. Cela peut être des

nombres qui s'actualisent, tout en restant lisibles (par exemple sauts de 5 en 5), une jauge de progression, ...

Interaction public :
- Défilement des bulles d'informations ;
- Passage à la page 3 avec la manette ;
- Possibilité de revoir l'animation une fois celle-ci terminée.

3. Règles du jeu

Éléments présents sur la page :
- Un titre ;
- Quelques lignes du texte, expliquant le contexte du jeu 1 (page 4) : *« Lorsque plusieurs personnes veulent rejoindre le même endroit en choisissant, chacune, le trajet le plus court, cela peut parfois créer des embouteillages. Pour éviter cela, on peut bien sûr adapter les routes : en rajouter ou en enlever, etc... Que choisirez-vous de faire ? »*.

Interaction public : lancer le jeu grâce à la manette.

4. Jeu 1

Le but du jeu ici est de trouver la configuration qui minimise le temps moyen de parcours pour relier A à D. Pour cela, la personne va devoir choisir quelle(s) route(s) elle ouvre, et quelle(s) route(s) elle ferme. Une fois le choix effectué, la personne lance la simulation.
La simulation reprend le principe expliqué pour l'animation de la page 2. Un compteur affiche le temps moyen. Une fois la simulation terminée, le score est enregistré.

Éléments présents sur la page :
- Un titre ;
- Une consigne : « Choisis quelles routes tu veux ouvrir et fermer. Quand tu as fini, lance la simulation. Objectif : faire passer 1000 voitures en minimisant le temps de parcours moyen. » ;
- Un cadre avec le plan des routes et le compteur du temps moyen.

Description de la simulation :
Le principe est le même qu'à la page 2, mais en adaptant le plan aux choix faits par la joueuse ou le joueur. On affiche le taux de saturation des routes, le nombre de voitures présentes sur chacune d'entre elles, et le temps moyen de parcours qui s'actualise au fur et à mesure jusqu'à la fin de la simulation. Une fois la simulation terminée, le score s'affiche en gros (le temps moyen obtenu) puis on passe automatiquement à la page 5.

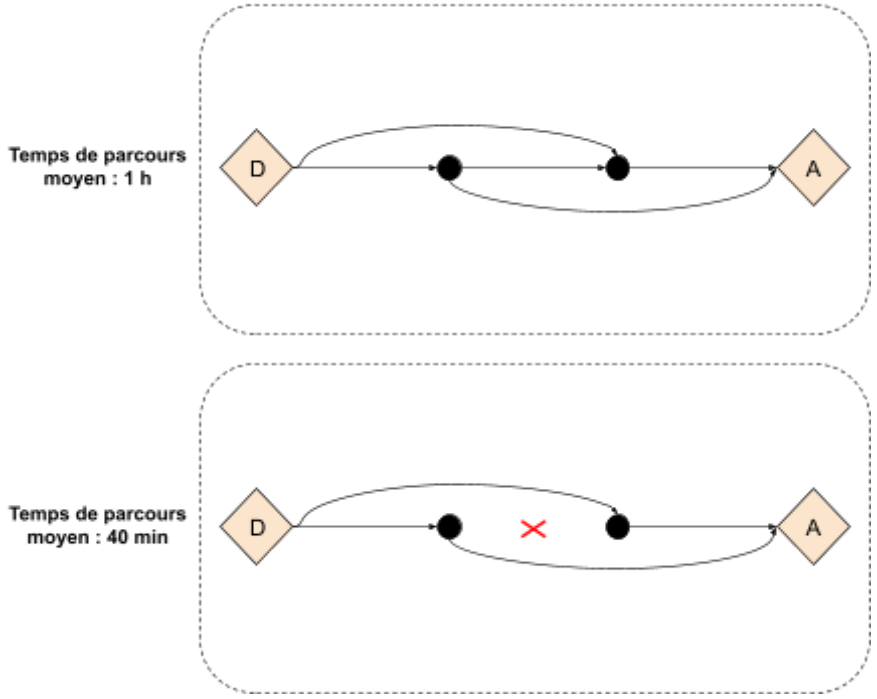
Interaction public :
- Ouvrir et fermer des routes grâce à la manette ;
- Lancer la simulation ;
- A partir de la deuxième tentative, la personne peut afficher la pop-up 4A (un indice).

4A. Indice

Cette pop-up peut être ouverte de deux façons. Soit la personne demande un indice d'elle-même à partir de la page 4. Soit la pop-up s'ouvre automatiquement, à partir de la 3ème partie du jeu 1.

Éléments présents sur la page

- Un titre ;
- Quelques lignes de texte ;
- Deux animations rapides, montrant la situation sur deux routes reprenant les plans ci-dessous :



Interaction public : fermer la pop-up pour revenir à la page 4.

5. Scores jeu 1

Éléments présents sur la page :

- Un titre ;
- Le tableau des scores (qui peut être pré-calculé à l'avance, puisqu'il y a un nombre réduit de possibilités) ;
- Le score que la personne vient d'obtenir et son classement.

Interaction public : la personne peut soit rejouer (retour page 4), soit passer à la suite (passage à la page 6).

6. Le paradoxe de Braess

Éléments présents sur la page

- Un titre ;
- Quelques lignes de texte, expliquant le paradoxe ;
- Une animation (la même que celle de la page 4A).

Interaction public : La personne peut soit passer au jeu 2 (passage à la page 7), soit revenir au jeu 1 (retour à la page 4), soit s'arrêter là dans la manipulation (passage à la page 10).

7. Transition vers jeu 2

Éléments présents sur la page :

- Un titre ;
- Quelques lignes de texte, introduisant la notion de régulation : on peut gérer la saturation des voies en ouvrant ou fermant dynamiquement des routes (comme avec des feux de circulation).

Interaction public : passage à la page 8.

8. Jeu 2

Le but du jeu est à nouveau de trouver la configuration qui minimise le temps moyen de parcours pour relier A à D. La différence avec le jeu 1 est que, cette fois, la simulation est dynamique : le public peut modifier l'ouverture / fermeture des routes *pendant* la simulation (un peu comme s'il gérait des feux de circulation).
La personne va donc devoir choisir quelle(s) route(s) elle ouvre, et quelle(s) route(s) elle ferme. Seules une ou deux routes peuvent être ouvertes ou fermées. La simulation s'actualise en temps réel. Le plan est beaucoup plus simple que pour le jeu 1.

La simulation reprend le principe expliqué pour l'animation de la page 2. Un compteur affiche le temps moyen. Une fois la simulation terminée, le score est enregistré et ajouté au tableau des scores. La personne peut alors choisir de rejouer ou non.

Éléments présents sur la page :

- Un titre ;
- Une consigne : « La route bleue peut être ouverte et fermée autant de fois que l'on veut pendant que les voitures empruntent le circuit. Objectif : minimiser le temps de parcours moyen. Pourras-tu faire le meilleur score ? » ;
- Un cadre avec le plan des routes et le compteur du temps moyen. La ou les routes pouvant être ouvertes / fermées devront être clairement identifiables.

Description de la simulation :
Le principe est le même qu'à la page 2, mais avec une adaptation supplémentaire de la simulation puisque le plan est modifié au cours de la simulation. On affiche le taux de saturation des routes, le nombre de voiture présentes sur chacune d'entre elles, et le temps moyen de parcours qui s'actualise en temps réel jusqu'à la fin de la simulation.

Interaction public :

- Lancer la simulation ;
- Ouvrir et fermer la ou les route(s) autant de fois que souhaité, pendant la simulation ;
- Une fois la simulation terminée, le score (c'est-à-dire le temps moyen obtenu) s'affiche en gros, puis on affiche automatiquement la page 8.

9. Scores jeu 2

Éléments présents sur la page :

- Un titre ;
- Le tableau des scores : les 10 meilleurs scores et le score qui vient d'être obtenu (avec le classement), à la façon des affichages des scores des jeux d'arcade ;
- Un espace pour choisir un avatar parmi une petite liste proposée ;
- Si la personne a fait un très bon score (podium, ou top 10) : une petite animation.

Interaction public :

La personne peut choisir son avatar et ajouter son score au tableau des scores. Celui-ci s'intercale alors au bon endroit, associé à l'avatar choisi.
Ensuite, la personne peut soit rejouer (retour page 7), soit terminer la manipulation (passage page 10).

Précisions pour le tableau des scores :

- Il faudra stocker le nombre total de joueurs et joueuses (pour afficher par exemple : « Tu es 100ème sur 568 »).
- Il faudra prévoir, pour les médiateurs et médiatrices, la possibilité de réinitialiser le tableau des scores. Il s'agirait d'un « faux » reset, puisqu'on garderait une dizaine de scores hardcodés dans l'application.

10. Fin

Une page très simple, avec un titre et un peu de texte, pour marquer la fin de la manipulation.

Application 3 – « Pique-nique »

L'histoire : *C'est bientôt l'heure du pique-nique ! Je cherche le meilleur endroit pour le repas, sans sortir du chemin ou revenir sur mes pas. Faut-il s'arrêter ici, ou bien y aura-t-il beaucoup mieux un peu plus loin ?*

Informations générales

Cette application sera intégrée à la « balade en montagne », au stand Montagne 3 du pré-programme muséographique. Le stand comprendra un panneau explicatif.

Matériel informatique

★ 2 tablettes de 13'' ou 14''. Celles-ci devront communiquer tout au long de la manipulation, car le jeu se fait en parallèle sur les deux tablettes en même temps.

Contexte scientifique

Le jeu présenté s'inspire d'un problème connu en mathématiques sous le nom de « problème des secrétaires », ou « problème du recrutement immédiat ». Le principe est le suivant : je dois choisir un élément (dans notre histoire, ce sont des lieux où pique-niquer) parmi une série de 10 (par exemple). Je ne sais pas quel est le « pire », ni quel est le « meilleur » élément, et n'ai aucune échelle de classement. Tous les candidats présentés sont différents, et leur ordre est aléatoire. ils sont présentés successivement : à chaque fois, je dois décider si je le garde ou non. Si je choisis de le garder, alors le jeu s'arrête. Si je décide de ne pas le garder, alors on me présente l'élément suivant. Il n'est pas possible de revenir en arrière. Le but du jeu est évidemment d'avoir le meilleur élément de la série à la fin.

Malgré le côté aléatoire du jeu, il existe une stratégie qui permet de maximiser la probabilité d'avoir le meilleur élément à la fin. Cette stratégie varie légèrement selon le nombre d'éléments de la série, mais elle consiste toujours à laisser passer une certaine quantité de candidats, sans jamais les choisir. Puis, une fois cette quantité passée, il faut prendre le premier qui est meilleur que tous les précédents.

Si nécessaire, l'ensemble des formules pour le calcul des probabilités sera fourni par la MMI.

Objectifs

★ Découvrir que, dans certains cas, on peut avoir une stratégie dans un jeu reposant sur le hasard.

★ Découvrir le dilemme « exploitation vs exploration » : vaut-il mieux choisir la meilleure option à partir de la connaissance partielle que l'on a déjà de la situation (exploitation), ou bien continuer à tester de nouvelles options en espérant qu'elles apportent une meilleure solution, et prendre le risque de « perdre » une solution que l'on a déjà ?

Vue générale de l'application

Dynamique de jeu

L'application est un jeu à deux personnes, qui s'affrontent pour obtenir le meilleur score. Les deux adversaires sont face à face, avec leur propre écran. Après le jeu, une partie explicative permet de rentrer plus en détail dans le contenu scientifique du stand.

Les règles du jeu sont les suivantes :

- On passe successivement devant 10 lieux de pique-nique, plus ou moins agréables. L'ordre dans lequel ils sont présentés aux joueurs et joueuses est aléatoire. En revanche, les deux adversaires visitent les lieux dans le même ordre.
- On ne peut pas revenir en arrière.
- Même si c'est invisible pour le public avant la fin de la partie, les lieux de pique-nique sont classés selon leur qualité et les uns par rapport aux autres. Aucun lieu n'est « aussi bien » qu'un autre, il y en a toujours un mieux que l'autre. Il s'agit bien d'un classement *relatif*. Il n'y a pas de « note » attribuée à chaque lieu, qui donnerait une idée de sa qualité indépendamment de tous les autres emplacements.
- A chaque lieu présenté, le joueur ou la joueuse a connaissance du classement des lieux déjà vus. Il ou elle doit alors choisir entre s'arrêter ou poursuivre pour voir les lieux suivants.
- Les deux meilleurs emplacements valent 1 point chacun. Tous les autres valent 0. Les personnes jouent plusieurs manches. La première personne à 2 points gagne, et le jeu s'arrête alors. S'il y a égalité, la première personne à marquer un point sans que l'autre ne marque gagne.

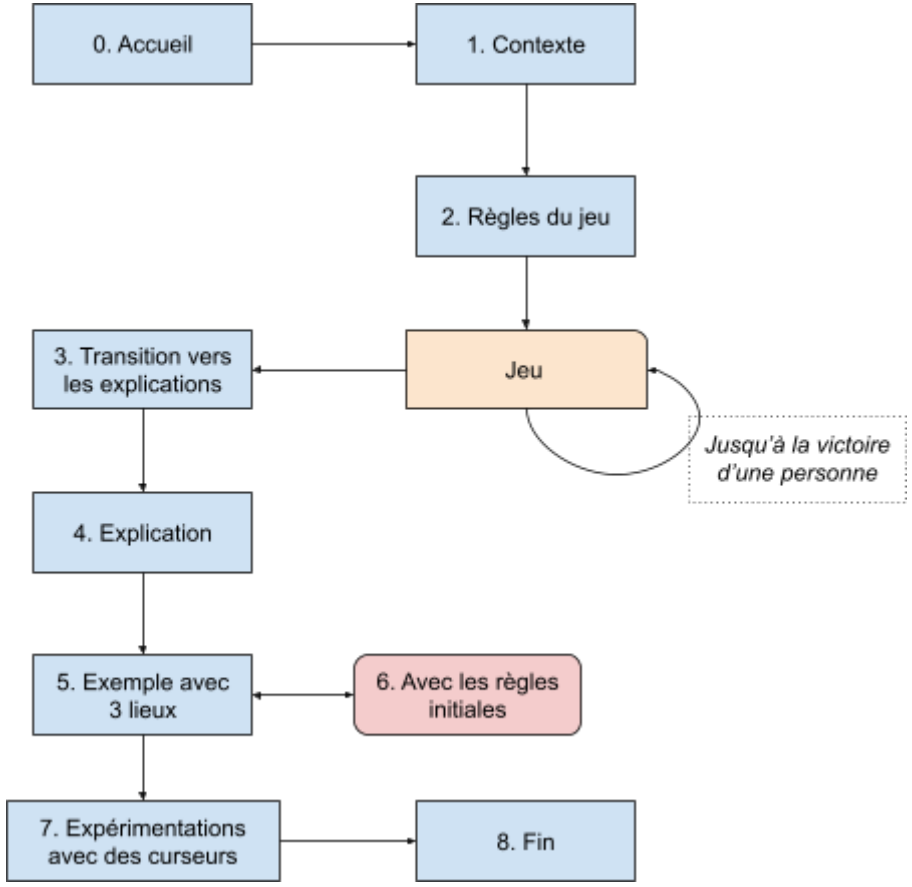
Précisions pour toute l'application

- Les textes ne sont pas définitifs.
- A tout moment, présence d'un bouton « Retour au début » qui renvoie à l'accueil.
- A tout moment, au bout d'un certain temps d'inactivité, on revient automatiquement à l'accueil.

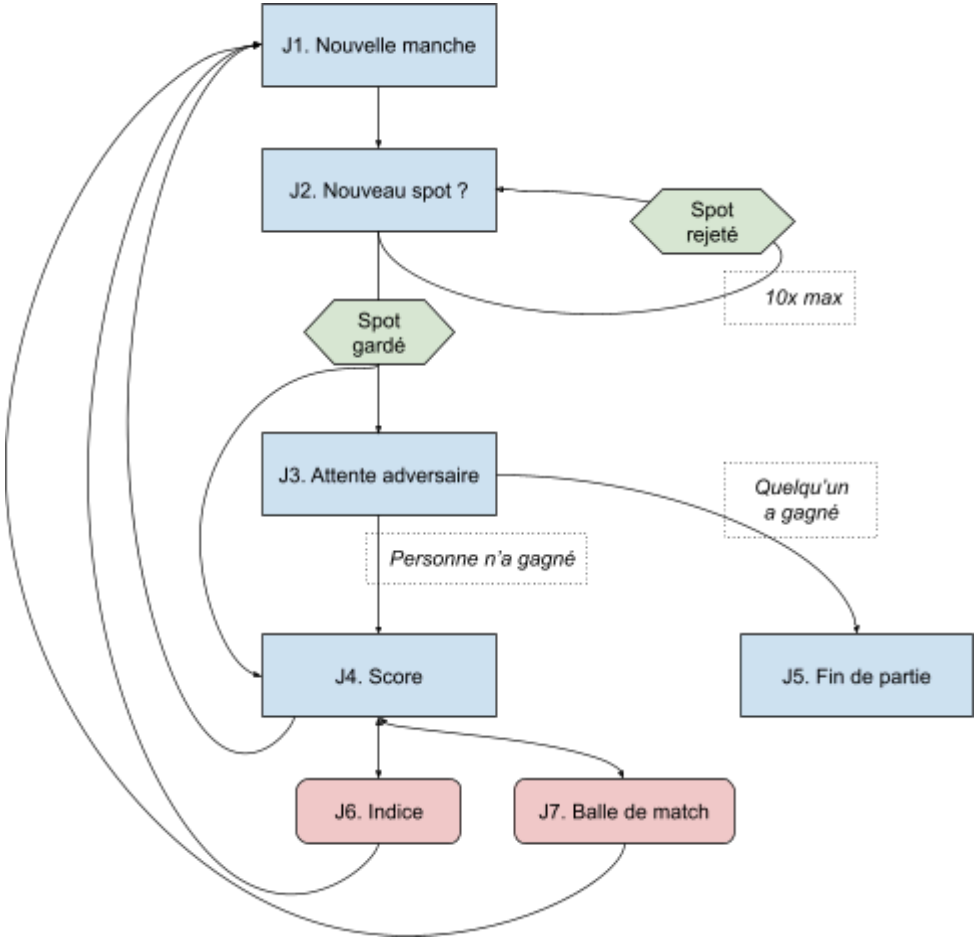
A charge de la MMI

- Le contenu scientifique des explications sera fourni par la MMI, incluant toutes les valeurs des probabilités de gain en fonction des stratégies adoptées, et les formules à utiliser pour le calcul de probabilités de la page 7.
- L'ensemble des textes de l'application seront rédigés par la MMI.

Scénario général de l'application



Scénario de la partie « Jeu » de l'application



Descriptif page par page

Scénario global

Dans toutes les pages du scénario global :

- Un bouton « Page précédente » permet de revenir en arrière ;
- Un bouton « Page suivante » permet d'avancer dans l'application.

0. Accueil

Une page d'accueil simple, avec le titre de l'application, un visuel et un bouton pour démarrer.

1. Contexte

Éléments présents sur la page :

- Un titre ;
- Quelques lignes de texte, pour expliquer le contexte, l'histoire et introduire le jeu ;
- Bouton « Médiation » qui permet de sauter l'étape du jeu et d'aller directement à la page 4. Cette action est protégée par un code.

Interaction public : pour les médiateurs et médiatrices, on peut passer à la page 4 après avoir saisi le code secret.

2. Règles du jeu

Éléments présents sur la page :

- Un titre ;
- Quelques lignes de texte expliquant les règles du jeu ;
- Éventuellement une illustration pour clarifier les règles ;
- Un bouton « Médiation » qui permet de sauter l'étape du jeu et d'aller directement à la page 4. Cette action est protégée par un code ;
- Un bouton « Lancer le jeu » (qui remplace le bouton « Suite »).

Interaction public :

- Pour les médiateurs et médiatrices, on peut passer à la page 4 après avoir saisi le code secret ;
- Passage au scénario « Jeu » avec le bouton « Lancer le jeu ».

3. Transition vers les explications

A partir de cette page et jusqu'à la fin (page 8), le contenu ne sera affiché que sur une seule tablette, afin de permettre aux deux personnes de découvrir les explications ensemble. Il faudra donc prévoir, sur la tablette inutilisée, une page fixe, invitant l'utilisateur ou l'utilisatrice à se déplacer vers l'autre écran. Cette page ne proposera aucune interaction. Elle repassera automatiquement à la page d'accueil lorsque l'écran « actif » fait de même.

Éléments présents sur la page :

- Un titre ;
- Quelques lignes de texte.

4. Explications

A partir de cette page, les explications autour de la stratégie sont données avec des règles du jeu légèrement simplifiées, permettant également de simplifier la stratégie. La règle utilisée pour les pages 4, 5 et 7 est la suivante : on gagne si l'on prend le meilleur lieu (il n'y a plus de cumul de points, et même le deuxième meilleur lieu ne permet pas la victoire).

Éléments présents sur la page :

- Un titre ;
- Plusieurs lignes de texte pour les explications, avec une ou deux illustrations pour clarifier l'explication.

5. Exemple avec 3 lieux

Éléments présents sur la page :

- Un titre ;
- Un court texte d'une ou deux lignes ;
- Un tableau à double-entrée (voir plus bas) ;
- Un bouton « La stratégie avec les règles initiales ».

Le tableau à double entrée contiendra les éléments suivants (les textes ne sont pas définitifs) :

Stratégie → _____ **Ordre d'apparition ↓**	**Stratégie 1 :** prendre le 1er	**Stratégie 2 :** prendre le 2ème	**Stratégie 3 :** prendre le dernier	**Stratégie 4 :** regarder le 1er, et prendre dès que c'est meilleur (ou prendre le dernier).
1 - 2 - 3	Gagné 🎉	Perdu 😢	Perdu 😢	Perdu 😢
1 - 3 - 2	Gagné 🎉	Perdu 😢	Perdu 😢	Perdu 😢
2 - 1 - 3	Perdu 😢	Gagné 🎉	Perdu 😢	Gagné 🎉
2 - 3 - 1	Perdu 😢	Perdu 😢	Gagné 🎉	Gagné 🎉
3 - 1 - 2	Perdu 😢	Gagné 🎉	Perdu 😢	Gagné 🎉
3 - 2 - 1	Perdu 😢	Perdu 😢	Gagné 🎉	Perdu 😢
Chances de gagner	**2 sur 6**	**2 sur 6**	**2 sur 6**	**3 sur 6**

Interaction public : le bouton « Avec les règles initiales » permet d'ouvrir la page 6 (sous forme de pop-up).

6. Avec les règles initiales

Éléments présents sur la page :

- Un titre ;
- Un texte d'explications de plusieurs lignes. Ce texte reprend l'explication donnée en page 4, mais avec les règles du jeu qu'ont expérimenté les deux personnes au début de l'application. La stratégie est plus compliquée, il s'agit donc d'un passage optionnel dans l'application ;
- Éventuellement une ou deux illustrations pour clarifier l'explication ;
- Un bouton « Fermer ».

Interaction public : fermeture de la pop-up grâce au bouton.

7. Expérimentation avec le curseur

Éléments présents sur la page :

- Un titre ;
- Une ou deux lignes de consigne ;
- 2 curseurs dynamiques :
    - Nombre de lieux à visiter ;
    - Temps d'observation (c'est-à-dire le nombre de lieux que l'on laisse passer sans rien prendre). Les valeurs du curseur « Temps d'observation » doivent changer en fonction du nombre de lieux à visiter (par exemple, pour 10 lieux, on peut avoir un temps d'observation compris entre 0 et 10 ; mais pour 100 lieux, il faut un temps d'observation compris entre 0 et 100).
- Un texte encadré dynamique, affichant le pourcentage de chances de gagner. Ce pourcentage dépend des valeurs choisies avec les curseurs.

Interaction public : on peut changer la valeur des curseurs, ce qui modifie le calcul de la probabilité de gagner.

8. Fin

Une page simple, avec un titre, une ligne de texte, et un bouton pour revenir à l'accueil.

Scénario jeu

Dans ces pages, il n'y a pas de bouton « Page suivante » ou « Page précédente » lorsque ce n'est pas précisé.

J1. Nouvelle manche

Éléments présents sur la page :

- Un titre ;
- Le rappel des scores des deux adversaires (avec les manches précédentes) ;
- Un bouton « Commencer le jeu ».

Interaction public : une fois que les deux adversaires ont cliqué sur « Commencer le jeu », la nouvelle manche se lance (passage à la page J2 qui présente le premier lieu de pique-nique possible).

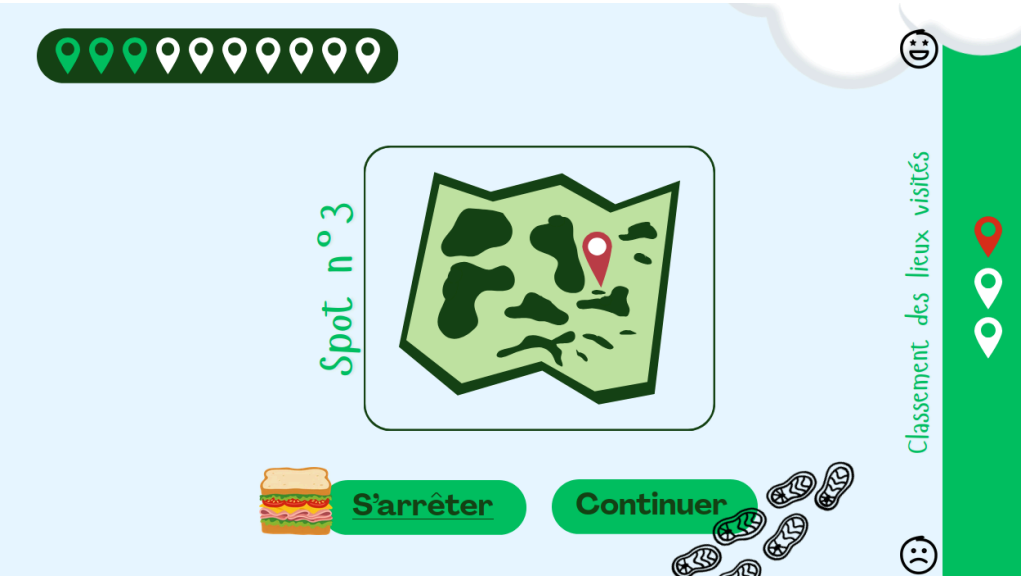
Remarque : à chaque nouvelle manche, l'ordre des lieux à visiter est de nouveau tiré au sort.

J2. Nouveau spot ?

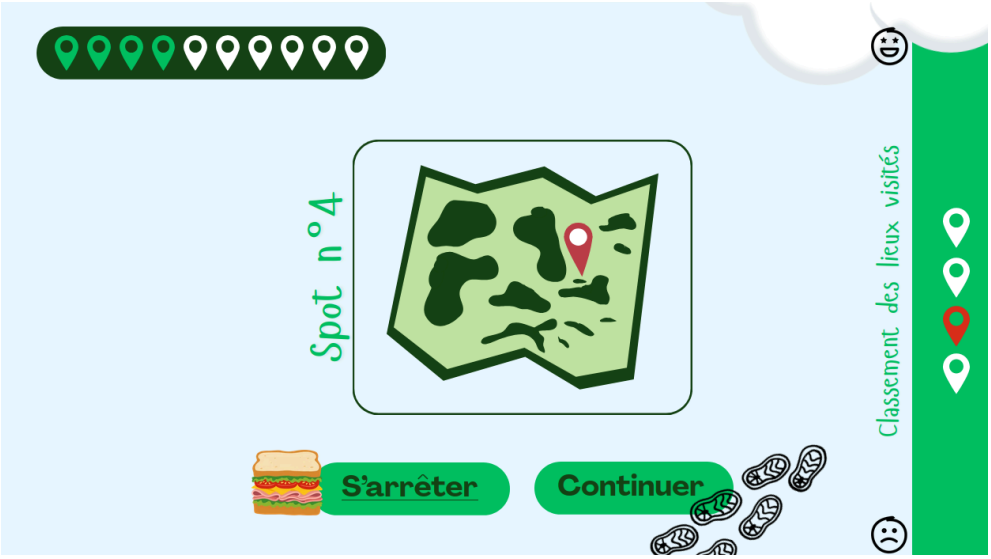
Éléments présents sur la page

- Un titre ;
- Une ligne de consigne ;
- Une indication du nombre de lieux restants à visiter ;
- Une illustration représentant le lieu actuel. Attention, cette illustration doit être neutre quant à la « qualité » du lieu visité (par exemple : un indicateur de position qui se déplace sur une carte) ;
- Une indication classant le lieu actuel par rapport aux lieux déjà visités (qui sont donc de plus en plus nombreux à mesure que l'on avance). Ce classement ne prend en compte que les lieux déjà visités (pas la totalité d'entre eux) ;
- Un bouton « S'arrêter ici » ;
- Un bouton « Continuer » (sauf pour le dixième lieu, le bouton n'est pas cliquable).

Pour clarifier la liste ci-dessus, voilà un exemple de ce à quoi pourrait ressembler ces pages. *Ces illustrations ne sont qu'indicatives, elles ne sont en rien une demande en termes de design final de l'application.*



Visite du 3ème lieu. C'est pour l'instant le meilleur des 3 visités.



Visite du 4ème lieu. Il est plutôt en bas de classement par rapport aux autres lieux déjà visités.

Indications sur la gestion des 2 écrans

Les deux adversaires jouent sur la même instance à chaque manche (c'est-à-dire les mêmes lieux, dans le même ordre).

Il est important que leur progression se fasse au même rythme. Ainsi, si l'une des personnes clique sur « Continuer » avant son adversaire, il faudra qu'elle attende que l'autre ait également fait son choix avant de voir la page suivante. On peut imaginer par exemple une petite pop-up qui s'affiche, indiquant qu'il faut patienter.

Interaction public

- Un clic sur le bouton « S'arrêter » peut soit :
    - Envoyer à la page J4 ou J5, dans le cas où l'adversaire s'arrête au même endroit, ou avait déjà choisi son emplacement ;
    - Envoyer à la page J3 sinon.
- Un clic sur le bouton « Continuer » : passage au lieu suivant.

J3. Attente adversaire

Cette page s'affiche sur l'une des tablettes quand un seul ou une seule des adversaires a choisi de s'arrêter, pour laisser à l'autre personne le temps de visiter les lieux suivants et de faire son choix. L'intérêt est de garder les deux adversaires au même niveau d'information dans le jeu, et de ne pas afficher le score alors que l'autre personne n'a pas encore terminé la manche. Il peut s'agir par exemple d'une page très simple avec un visuel et une ligne de

texte, ou bien des pages similaires à J2, qui s'affichent au fil de la progression de l'adversaire, mais où plus aucune interaction n'est possible.

Il n'y a pas d'interaction à prévoir pour cette page. Une fois que la deuxième personne a choisi son lieu, on passe automatiquement à la page J4 (si personne n'a gagné) ou J5 (si quelqu'un a gagné).

J4. Scores

Eléments présents sur la page

- Un titre ;
- Le score des deux adversaires ;
- Un bouton « Nouvelle manche ».

Pop-up conditionnelles

- Si les scores n'ont pas changé depuis 3 parties : affichage automatique de J6, après quelques secondes (pour laisser le temps de voir les scores). Dans ce cas, le bouton « Nouvelle manche » ne devient cliquable qu'à la fermeture de la pop-up.
- Si les deux personnes atteignent en même temps le score 2 points : affichage automatique de J7, après quelques secondes (pour laisser le temps de voir les scores). Dans ce cas, le bouton « Nouvelle manche » ne devient cliquable qu'à la fermeture de la pop-up.

Interaction public : le bouton « Nouvelle manche », envoie sur la page J1 avec une nouvelle série de lieux.

J5. Fin de partie

Cette page ne s'affiche qu'une fois que l'un ou l'une des deux adversaires a gagné.

Éléments présents sur la page

- Un titre ;
- Les scores des deux adversaires ;
- Quelques lignes de texte (différent sur les deux écrans, selon qui a gagné et perdu) ;
- Un bouton « Suite ».

Interaction public : le bouton « Suite » envoie à la page 3 du scénario global.

J6. Indice

Cette page permet de guider les adversaires vers la bonne stratégie, afin que la partie ne dure pas trop longtemps. Il s'agit d'une pop-up qui s'ouvre automatiquement.

Éléments présents sur la page

- Un titre ;
- Quelques lignes de texte ;
- Un bouton « Fermer la fenêtre » ;
- Un bouton « Nouvelle manche ».

Interaction public

- Fermeture de la pop-up pour revenir à la page J4 ;
- Lancement d'une nouvelle manche (page J1) avec le bouton « Nouvelle manche ».

J7. Balle de match

Cette page s'affiche dans le cas d'une égalité à deux points partout, pour indiquer (ou rappeler) la règle qui s'applique dans ce cas particulier.

Éléments présents sur la page

- Un titre ;
- Un texte de quelques lignes ;
- Un bouton « Fermer », qui apparaît au bout de quelques secondes ;
- Un bouton « Nouvelle manche », qui apparaît au bout de quelques secondes.

Interaction public

- Fermeture de la pop-up pour revenir à la page J4 ;
- Lancement d'une nouvelle manche (page J1) avec le bouton « Nouvelle manche ».

Application 4 – « Les lucioles »

L'histoire : *Je pars à la découverte d'un endroit où des lucioles ont la réputation d'avoir un comportement presque magique. Elles clignotent en rythme, et de façon synchronisée.*

Informations générales

Cette application sera intégrée à la « balade de nuit », au stand Nuit 1 du pré-programme muséographique. L'application sera découpée entre une tablette (avec laquelle le public sera en interaction) et une partie projetée.

Matériel informatique

- ★ 1 tablette de 13'' ou 14'' *OU* un écran et ordinateur ;
- ★ Un vidéoprojecteur *(qui n'est pas à la charge de l'équipe de développement car inclus dans le lot « Scénographie »)*.

Contexte scientifique

Certaines rares espèces de lucioles, vivant en Asie et en Amérique du Nord, ont la particularité de synchroniser leurs clignotements lorsque certaines conditions sont réunies (population suffisante, positionnement adéquat des individus...). Plusieurs vidéos, comme celle-ci de National Geographic, montrent le phénomène observé dans la nature. L'idée générale est que les lucioles vont clignoter individuellement à un certain rythme, toujours le même. Lorsque l'une d'entre elle « flashe », elle va influencer ses voisines (si elles sont suffisamment proches), qui vont faire un petit « saut » dans leur cycle. Leur clignotement arrivera donc plus tôt que prévu. Ainsi, petit à petit, un groupe peut finir par se synchroniser parfaitement.

Cela peut être modélisé avec des outils mathématiques, en utilisant la notion d'oscillateurs couplés. Dans la suite de l'explication et l'ensemble de la manipulation, tous les oscillateurs ont la même fréquence. On peut voir un système d'oscillateurs couplés comme un graphe, où chaque nœud a une phase donnée, et influence la phase de ses voisins. Un exemple connu pour illustrer cela est de poser sur une planche mobile plusieurs métronomes mécaniques battant au même rythme, mais avec une phase décalée. Les métronomes, en faisant bouger légèrement la planche, vont petit à petit se décaler pour finir par battre de façon synchronisée, comme sur cette vidéo. Ceci est un exemple de synchronisation de phases, assez proche du phénomène observé chez les lucioles.

Dans le cas des lucioles, plusieurs paramètres vont influencer sur la synchronisation du groupe (est-ce qu'elle finit par arriver, et au bout de combien de temps) : la densité de la population, la distance à laquelle elles doivent être pour s'influencer, la puissance avec

laquelle elles s'influencent (c'est-à-dire à quel point elles décalent le cycle de leurs voisines lorsqu'elles flashent), etc. Pour les simulations, nous utiliserons le modèle de Mirollo-Strogatz.

Comportement et interaction des lucioles du modèle

Le comportement lucioles du modèle est donné pour comprendre l'algorithme à écrire. Il n'est pas nécessaire de tout comprendre parfaitement à ce stade : tout pourra ensuite être discuté et explicité avec la MMI. Ces informations ont pour but de clarifier l'algorithme demandé, et d'aider à évaluer la demande.

On va travailler sur un tore (un carré de taille L dans lequel on recolle la droite et la gauche et le bas et le haut). Les lucioles seront donc dans des positions (x,y) avec par exemple $0 \leq x \leq 10$ et $0 \leq y \leq 10$ mais la position (5,10) est la même que (5,0) : une luciole qui sort par la droite du carré re-rentre par la gauche. Cela simplifie les interactions entre les lucioles, simplifie l'algorithme et accélère la synchronisation. Comme ce n'est pas très réaliste, on ne projettera qu'une partie du carré pour que cette astuce ne se voit pas, par exemple le carré $1 \leq x \leq 9$, $1 \leq y \leq 9$. Si besoin, le calcul de distance entre deux lucioles dans ce cadre sera explicité par la MMI.

Les paramètres de la simulation

Tous les paramètres seront fournis et/ou ajustés après test par la MMI. Dans la suite, les paramètres surlignés en vert sont ceux qui peuvent être modifiés par le public.

- La taille du carré / tore (**L**) et la taille du carré projeté.
- Agitation des lucioles : **A**. Celle-ci pourra être choisie à l'aide d'un curseur parmi quelques choix prédéterminés par le public selon le scénario ci-dessous. Mettre ce paramètre à 0 sera accepté, ce qui rend les lucioles fixes. En fonction de la valeur de ce paramètre, c'est le mouvement des lucioles, indépendants de l'algorithme d'influence entre les lucioles, qui change.
- Force d'interaction : **F**. C'est un paramètre qui, lorsqu'il grandit, rend la synchronisation plus rapide. Plus F est grand, plus une luciole qui flashe influence fortement ses voisines. Ce paramètre pourra être choisi par le public, toujours parmi quelques choix pré-déterminés (4 ou 5).
- Distance d'influence : **D**. Il s'agit de la distance d'interaction des lucioles. Une luciole n'influence que celles qui sont à distance plus petite que D d'elle lorsqu'elle flashe. Ce paramètre pourra être choisi par le public, toujours parmi quelques choix pré-déterminés (4 ou 5). Plus D est grand, plus la synchronisation est rapide.
- Nombre de lucioles : **N**. Ce paramètre sera figé et déterminé par la MMI.

- Temps de flash : **C** (pour clignotement). Cela correspond à la durée du flash d'une luciole (par exemple 60 ms). Ce paramètre sera figé et déterminé par la MMI en fonction du rendu esthétique de l'application.
- Fréquence de flash : **T_0.** Cela correspond au temps que passe une luciole, sans influence, entre deux flashs.

Dans la suite, pour simplifier le discours, on va considérer T_0 = 1 s =1000 ms et on prendra un pas de temps p (pour les calculs) de 1 ms. Le temps de flash sera C = 60 ms. Tout ceci sera ajusté et choisi par la MMI après des tests.

Le mouvement des lucioles

A chaque pas de temps, chaque luciole a une position (x,y) et une direction de vitesse (donnée par un angle Θ).
En fonction du paramètre A, on aura une vitesse V, un paramètre Θ_0 de changement de direction et un temps T_1 de fréquence de changement de direction. La détermination de ces paramètres en fonction des choix du public (A = 1,2,3,4,...) sera donnée par la MMI.
Les positions et vitesses des lucioles au départ seront soit choisies au hasard, soit pré-déterminées selon le scénario ci-dessous.

Une fois muni de ce paramètre A et donc de la vitesse V et des paramètre Θ_0 et T_1, à chaque pas de temps :

- La position de la luciole est actualisée : $(x, y) \rightarrow (x + v \times cos(\Theta) \times p,\ y + v \times sin(\Theta) \times p)$ où p est le pas de temps. Si la luciole sort du carré après actualisation, par exemple si x≥10, il faudra la remettre dans le carré (x → x-10).
- Tous les T_1 pas de temps, on actualise la direction : Θ → Θ + un angle random entre $-\Theta_0$ et Θ_0. Plus Θ_0 est grand, plus les lucioles peuvent changer de direction brusquement. Plus V est grand, plus elles se déplacent vite. Les lucioles suivent donc un mouvement aléatoire, plus ou moins agité.

Les horloges des lucioles

Chaque luciole possède une horloge, un entier compris entre 0 et 999 (ici parce qu'on a pris T_0 = 1 s = 1000 ms pour toutes les lucioles et un pas de temps d'1 ms).

- A chaque pas de temps, en dehors des interactions explicitées ci-dessous, on incrémente l'horloge de 1.
- Si son horloge arrive à 1000, elle retombe à 0.
- Si son horloge est entre 0 et C = 60 (la durée du flash), la luciole est affichée colorée. Le graphisme et l'animation du flash sont à la charge du prestataire.

Interaction entre les lucioles

A chaque pas de temps, un certain nombre de lucioles ont leur horloge qui arrive à 0. Souvent aucune ou une seule, puis de plus en plus et en des temps de plus en plus espacés à mesure qu'elles se synchronisent.

On a donc une liste de lucioles L qui viennent d'arriver à 0 après l'incrément de 1. Celles-ci vont influencer leurs voisines.

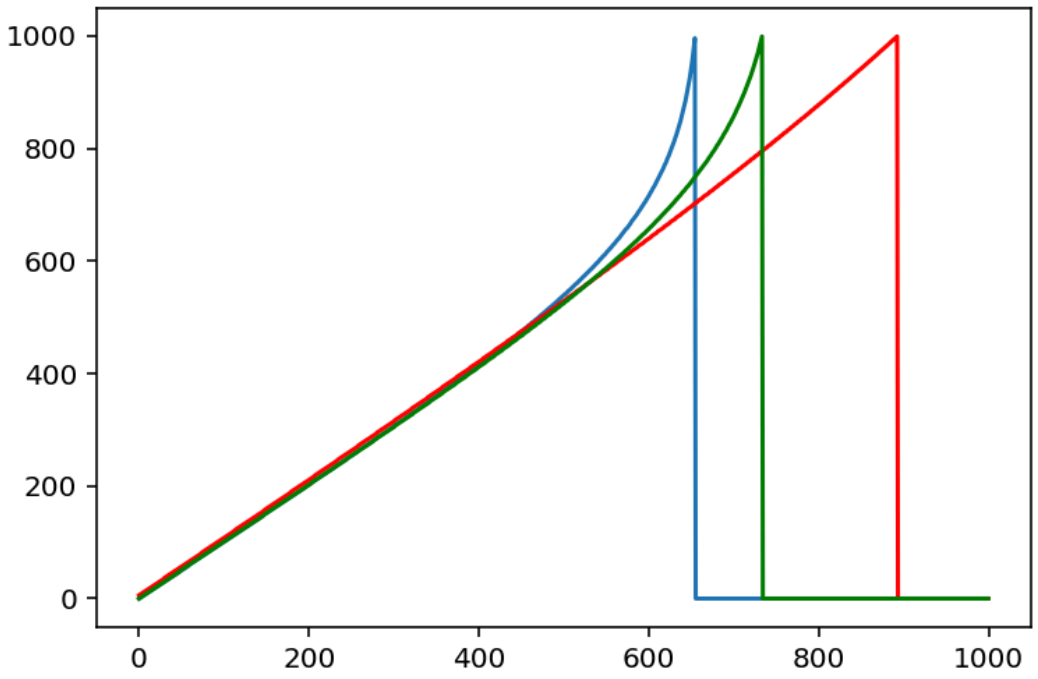
Le paramètre F de force d'interaction (qui n'aura que quelques positions possibles) déterminera deux paramètres f et g. Ces paramètres seront donnés par la MMI.

Pour toute luciole L_1 de la liste L :

- Pour toute luciole L_2 (entre 1 et N), si la distance entre L_1 et L_2 est plus petite que D (le paramètre de distance d'influence), l'horloge de L_2 va être mise à jour ainsi :
  Si $horloge\ (L_2) \geq\ -\ \frac{ln(f+(1-f)exp(-gT_0))}{g}$ ou $horloge\ (L_2)\ =\ 0$, alors $horloge\ (L_2) \rightarrow 0$.
  Sinon, $horloge\ (L_2)\ \rightarrow -\ \frac{ln(exp(-g\times horloge(L_2))-f\times(1-exp(-gT_0)))}{g}$ arrondi à l'entier inférieur.

Ici, T_0 = 1000. A noter, avec cette actualisation, une horloge à 0 reste à 0, une horloge proche de 1000 passe à 0 et l'incrément est d'autant plus important que l'horloge était proche de 1000.

Voici un exemple de graphes de cette actualisation :



Ces graphes représentent la fonction décrite ci-dessus : en abscisse, horloge(L_2) avant interaction ; en ordonnée, horloge(L_2) après interaction. En rouge, l'interaction est faible. En bleu, elle est très forte.
Comment lire ce graphique ? Si une luciole L_2 a une horloge à 800 au moment où sa voisine L_1 flashe, l'horloge de L_2 passe à 0 (et la luciole flashe) dans les cas vert et bleu, et à 840 dans le cas rouge.

- Remarque : une luciole peut subir plusieurs incréments parce que plusieurs lucioles de la liste L sont proches d'elle. Lors de cette étape de synchronisation, on n'a pas besoin de se soucier de l'état des lucioles : une luciole L_2 qui a son horloge à 0 voit son horloge rester à 0 par la formule ci-dessus. On peut donc appliquer le changement à toutes les lucioles voisines de L_1 indépendamment de leur état, pour toutes les L_1 dans la liste L.
- Attention : la liste des lucioles qui flashent a été établie à la fin d'un pas de temps, avant synchronisation. Les nouvelles lucioles arrivant à 0 suite à cette synchronisation n'influencent pas leurs voisines à ce tour. Il faudra attendre qu'elles fassent un tour complet. Il n'y a pas de cascade d'influence : ce n'est pas parce que horloge(L_2) passe à 0 qu'elle influence les autres, si elle n'était pas déjà dans la liste L.

Algorithme résumé

A chaque itération de temps T :

- Actualisation des positions des lucioles et actualisation des directions si T est un multiple de T_1. Voir *Le mouvement des lucioles*.
- Actualisation des horloges des lucioles : +1. Voir *Les horloges des lucioles.*
- On met dans une liste L toutes les lucioles dont l'horloge est arrivée à 0.
- On synchronise les lucioles comme expliqué dans *Interaction entre les lucioles*, ce qui actualise les horloges de certaines lucioles.
- Affichage des lucioles si T est un multiple d'affichage. Par exemple, si on a un affichage de 50 frames / seconde, si T est un multiple de 1000/50=20. Les lucioles sont éclairées si 0 ≤ horloge(luciole) ≤ C=60.
- T → T+1.

Et on boucle.

Objectifs

★ Découvrir un exemple de modélisation mathématique d'un phénomène naturel.
★ Découvrir que le comportement d'un individu (ici, une luciole) peut être très simple à comprendre et à modéliser, mais que cela devient très compliqué dès qu'on essaie de comprendre le phénomène d'ensemble.
★ Passer un bon moment à observer ce qu'il se passe, éveiller sa curiosité.

Vue générale de l'application

Dynamique de jeu

L'application a pour sujet la modélisation mathématique de certaines populations de lucioles qui ont la particularité de synchroniser leurs clignotements. Le public va donc être amené à voir des lucioles clignoter en se synchronisant ou pas, dépendant de certains paramètres.
L'application est divisée sur deux supports : une tablette par laquelle passent toutes les interactions avec le public (qui sert de « poste de contrôle » en quelque sorte) ; et une projection « en grand » des explications et des simulations avec les lucioles.

La manipulation navigue volontairement entre :

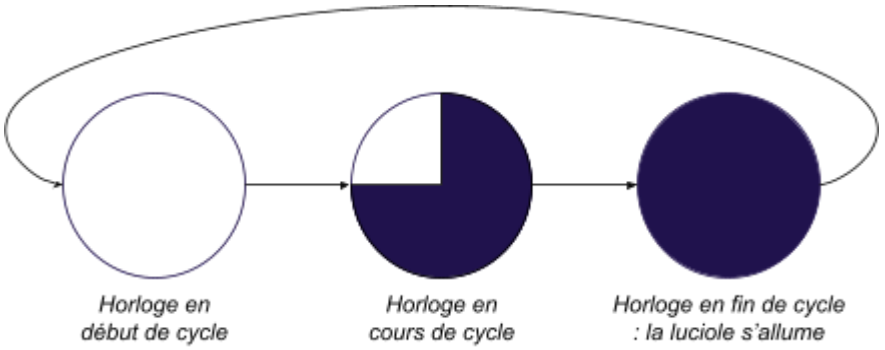
- **Des parties assez contemplatives**, où il n'est pas forcément nécessaire de tout comprendre, mais où l'on fait appel au côté merveilleux, magique du phénomène. Il n'est pas grave que des personnes s'y perdent un peu et y passent plusieurs minutes, en observant les animations sans forcément être actif ou active.

- **Des parties plus didactiques,** où l'objectif est de faire comprendre la modélisation et quelques-uns des paramètres qui la composent.

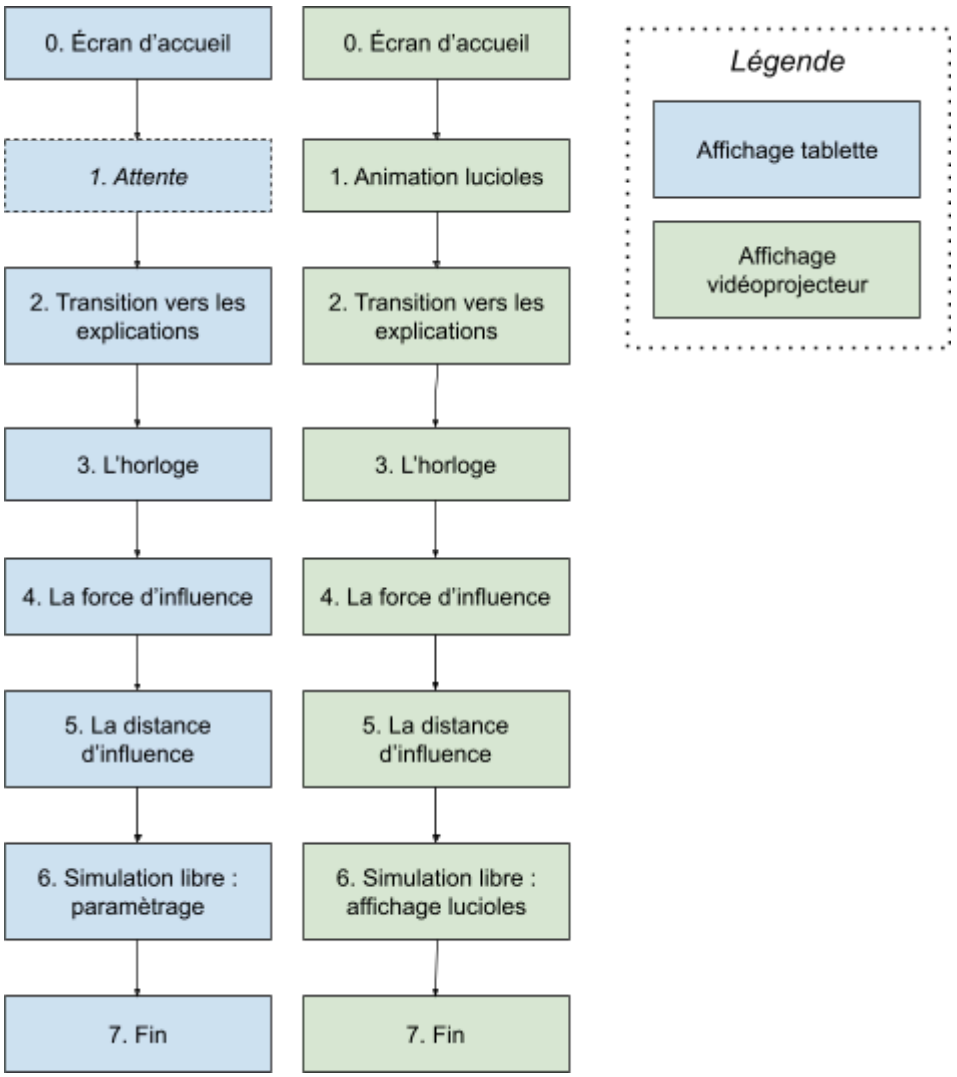
Affichage des lucioles

Les lucioles seront représentées de deux façons différentes dans l'application :

- **Format « classique »**, avec l'apparence d'une luciole. C'est l'affichage par défaut.
- **Format « horloge »**, nécessaire pour comprendre certains phénomènes (la fréquence, la puissance d'influence). Cet affichage est précisé quand il est souhaité. Il prend la forme d'une jauge circulaire, qui se remplit progressivement (la vitesse de remplissage dépend de T_0). La luciole s'allume lorsque la jauge est remplie. Celle-ci se vide ensuite d'un coup pour reprendre son cycle depuis le début.



Scénario



Précisions pour toute l'application

- Les textes et schémas ne sont pas définitifs.
- A tout moment, présence d'un bouton « Retour au début » qui renvoie à l'accueil.
- A tout moment, au bout d'un certain temps d'inactivité, on revient automatiquement à l'accueil.
- Pour tous les curseurs, un petit nombre de valeurs fixes seront choisies en avance, pour limiter les possibilités de la simulation.

A la charge de la MMI

- Les textes ;
- La position des lucioles au début de la simulation libre ;

- Les valeurs de tous les paramètres modifiables par le public (curseurs, cases à cocher) ;
- L'algorithme précis suivi par les lucioles ;
- Le contenu scientifique concernant les simulations.

Descriptif page par page

0. Accueil

Écran fixe, avec un bouton pour lancer l'application.

1. Animation lucioles

Écran de la tablette	Projection
/	Une animation de 20 secondes maximum, représentant un nuage de lucioles qui clignotent, aléatoirement au début, puis qui finissent par se synchroniser. Cette animation sera faite avec le même algorithme et les mêmes visuels que la page 6.

2. Transition vers les explications

Écran de la tablette	Projection
Éléments à afficher Bouton « Suite ». *Interaction public* Passage à la page suivante.	*Éléments à afficher* Un titre. Quelques lignes de texte, pour introduire la situation et les explications qui vont suivre.

3. L'horloge

Écran de la tablette	Projection
Éléments à afficher Bouton « Suite ». *Interaction public* Passage à la page suivante.	*Éléments à afficher* Un titre. Quelques lignes d'explication. Une animation : la luciole au format horloge. On voit l'horloge qui se remplit, et la luciole qui s'allume.

Exemple d'explication : « Les lucioles clignotent toutes à la même vitesse, à intervalle régulier. Comme si elles avaient une petite horloge interne, avec une aiguille qui tourne toujours à la même vitesse. Quand l'aiguille arrive en haut, la luciole s'allume. Et ça recommence. »

4. La force d'influence

Écran de la tablette	Projection
Éléments à afficher Un titre. Une consigne pour accompagner le curseur. Un curseur permettant de régler la force d'influence des lucioles. Un bouton suite. *Interactions* Quand la personne modifie la valeur du curseur, la force d'influence change (plus la valeur est haute, plus elles décalent fortement le cycle de leur voisine). Passage à page suivante avec le bouton.	*Éléments à afficher* Un titre. Quelques lignes d'explication. Deux lucioles (immobiles), accompagnées de leur horloge, qui clignotent. Elles s'influencent l'une l'autre au moment où elles s'allument, avec une force définie par le curseur. *Interaction* Quand la personne modifie la valeur du curseur, la force d'influence change (plus la valeur est haute, plus elles décalent fortement le cycle de leur voisine).

Exemple d'explication pour la vidéoprojection : « Une luciole qui s'allume influence ses voisines : l'aiguille de leur horloge va avancer d'un coup. Si elle est suffisamment avancée dans son cycle, une voisine peut même s'allumer immédiatement alors que ça n'était pas prévu ! »

Exemple de consigne pour la tablette : « Fais varier la force d'influence que les lucioles ont les unes sur les autres, et observe ce qu'il se passe. »

5. La distance d'influence

Écran de la tablette	Projection
Éléments à afficher Un titre. Une consigne pour accompagner le curseur. Un curseur permettant de faire varier la distance d'influence, et le diamètre du cercle entourant la luciole centrale. Un bouton suite. *Interaction public* Quand la personne modifie la valeur du curseur, la distance d'influence change. Celle-ci est illustrée par le diamètre du cercle, qui change donc également. Les lucioles qui sont suffisamment proches s'influencent les unes les autres. Passage à page suivante avec le bouton.	*Éléments à afficher* Un titre. Quelques lignes d'explication. Trois lucioles (immobiles), accompagnées de leur horloge, qui clignotent. La luciole centrale est entourée d'un cercle dont elle est le centre. Deux lucioles dans le même cercle s'influencent l'une l'autre au moment où elles clignotent, mais la force d'influence est figée. *Interaction public* Quand la personne modifie la valeur du curseur, le diamètre du cercle change. Les lucioles qui se trouvent dans le cercle s'influencent les unes les autres.

6. Simulation libre

Écran de la tablette	Projection
Éléments à afficher Curseurs : - Distance d'influence - Force d'influence - Agitation Bouton « Terminer ». Bouton « Réinitialiser les clignotements ». Bouton « Tout réinitialiser ». *Interaction public* La personne peut modifier les différents curseurs. Lorsqu'un paramètre est modifié, cela actualise la simulation (à la volée). Le bouton « Réinitialiser les clignotements » réinitialise au hasard les horloges de toutes les lucioles, sans modifier les paramètres. Le bouton « Tout réinitialiser » arrête la simulation et revient à l'affichage initial décrit ci-dessous. Le bouton « Terminer » envoie à la page 7.	*Éléments à afficher* Simulation des lucioles. *Interaction public* La simulation s'actualise quand la personne modifie un paramètre sur la tablette.

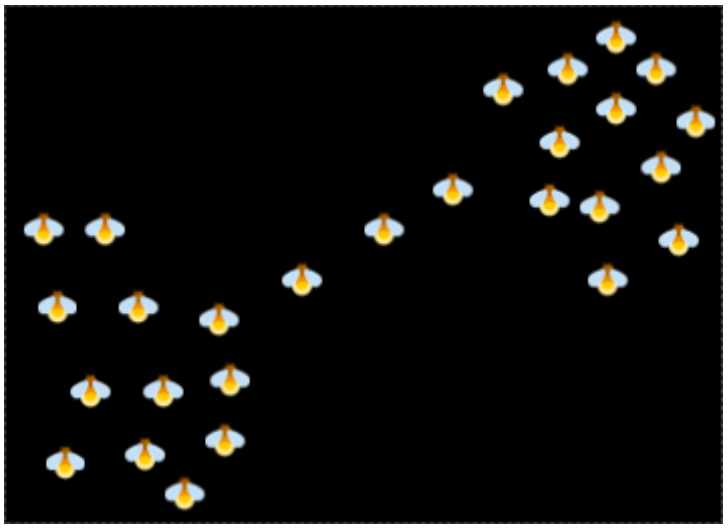
Affichage initial (à l'arrivée sur la page, et lors de l'appui sur « Tout réinitialiser ») :

Dans l'affichage initial, les lucioles sont immobiles et éteintes. Elles sont positionnées selon un graphe ressemblant au schéma ci-dessous (non-définitif). Un bouton « Lancer la simulation », en gros sur la tablette, permet de commencer la simulation avec les paramètres définis par défaut, en initialisant les horloges de chaque luciole au hasard. Une fois la simulation commencée, le bouton de lancement s'efface et laisse place aux curseurs et boutons décrits plus haut.

Dans l'affichage initial, les lucioles se comportent comme si elles étaient sur un plan, c'est-à-dire que les lucioles tout à gauche de l'écran n'influencent pas celles qui se trouvent à droite de celui-ci (et de même pour haut / bas).

Dès que l'on sort de l'affichage initial, les lucioles se comportent comme si elles étaient sur un tore, c'est-à-dire comme si l'écran était replié sur lui-même : les lucioles à gauche de

l'écran influencent celles de droite, celles du haut influencent celles du bas et, dans le cas où elles se déplacent, elles peuvent disparaître à une extrémité pour réapparaître en face.



Précisions sur les curseurs et les paramètres :

- *Force d'influence* – le nombre de niveaux sera restreint (environ 5 possibilités). Le détail de chacun sera fourni par la MMI.
- *Distance d'influence* – le nombre de niveaux sera restreint (environ 5 possibilités). Le détail de chacun sera fourni par la MMI.
- *Agitation* – ce curseur permet de gérer le déplacement des lucioles. Il ne propose que 3 niveaux :
    - lucioles immobiles ;
    - agitation faible : les lucioles avancent lentement. Lorsqu'elles clignotent, elles changent aléatoirement de direction parmi un nombre restreint de possibilités ;
    - agitation forte : les lucioles avancent plus rapidement. Lorsqu'elles clignotent, elles changent aléatoirement de direction parmi un nombre plus grand de possibilités (par exemple, elles peuvent faire demi-tour, ce qui n'était pas possible en agitation faible).

7. Fin

Un écran fixe, avec un peu de texte pour conclure la manipulation.