

SCS – Projet de session

Le cours SCS ne demandera de vous qu'un seul travail pratique pour la session entière. Ce travail sera découpé en livrables qui s'assembleront les uns dans les autres en cours de route.

Problématique

L'Agence spatiale canadienne (ASC) œuvre en préparation d'éventuels vols humains vers la planète Mars. Ce type de vol pose plusieurs questions, que ce soit sur le plan biologique (gestion de l'eau, de la nourriture, de l'oxygène, des maladies, de la porosité des ossatures en gravité restreinte, etc.), sur le plan technologique (alimentation électrique à long terme, mises à jour logicielles à distance, gestion des pannes, etc.) ou sur le plan psychologique (gestion des émotions, coexistence en milieu fermé, absence prolongée des proches, etc.).

En cette période où l'intelligence artificielle a la cote, l'ASC souhaite faire un peu d'impartition et cherche un tiers parti capable de développer une infrastructure qui permettra de tester différents modèles de comportement possibles, ceci dans le but de mieux préparer ces périlleuses (mais excitantes!) missions.

L'ASC possède sa propre équipe de spécialistes en intelligence artificielle, mais ses spécialistes de développement de systèmes répartis sont sollicités à temps complet par d'autres projets, ce qui explique que votre petite entreprise ait été mise au fait d'un appel de candidatures.

Votre équipe et vous-même souhaitez bien sûr obtenir ce contrat qui pourrait donner de l'élan à votre entreprise. Après tout, les systèmes répartis, c'est votre expertise!

Intention

L'idée générale derrière le projet est de trouver le moyen de simuler plusieurs modèles de comportement dans un environnement clos, avec un nombre arbitrairement grand de protagonistes. Il faut être capables d'utiliser plusieurs environnements, un seul par simulation, et il faut être capable d'utiliser plusieurs modèles de comportement dans une même simulation.

Le projet est de mettre en place une preuve de concept, qui pourra devenir un outil plus élaboré dans le futur. Pour cette première étape qu'est l'appel d'offres à laquelle vous participez, ce qui est visé est un simulateur de souris dans un labyrinthe.

Le logiciel devra être un système réparti contenant au moins un client, pour contrôler les paramètres de la simulation, visualiser celle-ci pendant son exécution, et visualiser les résultats. Cela dit, il est souhaitable que plusieurs autres composants participent à ce système. Par exemple, l'ASC envisage :

- Un serveur de labyrinthes
- Des composants pour contrôler l'intelligence artificielle des souris
- Un composant modélisant de règles pour évaluer l'évolution de la simulation et le bien-être physique comme psychologique des souris, etc.

Les souris auront une intelligence simulée, un objectif à atteindre, un état physique (entier dans l'intervalle $[0,10]$, où 0 indique un décès et 10 indique une parfaite santé) et un état psychologique (entier dans l'intervalle $[0,10]$, où 0 indique un état dépressif et 10 indique un état de bonheur et d'accomplissement total).

Ce que l'ASC souhaite pouvoir faire à partir d'un client est :

- Choisir un labyrinthe parmi ceux disponibles
- Choisir un nombre de souris pour la simulation (il sera idéalement possible d'avoir un nombre arbitrairement grand de souris, mais il qu'une simulation soit au moins capable d'accepter une, deux ou trois souris)
- Associer à chaque souris une intelligence comportementale parmi celles disponibles
- Associer à la simulation un ensemble des règles parmi celles disponibles
- Associer un ou plusieurs objectifs (des morceaux de fromage, par exemple) au labyrinthe
- Lancer la simulation
- Lorsque la simulation est terminée, être en mesure d'examiner les résultats

L'ASC serait d'accord pour qu'il soit possible pour des clients de visualiser la simulation, sans plus, tant qu'il est possible de faire l'ensemble du travail sur au moins un poste client.

Les scientifiques et les ingénieurs de l'ASC ont accès à plusieurs types d'équipement, et sont disposées à recevoir des offres utilisant *Windows 10*, *Linux*, *MacOS*, etc. du côté serveur. Puisque les usagers des processus clients ont des postes *Windows 10*, le code client doit absolument fonctionner au moins sur cette plateforme¹.

Le monde étant en évolution, les usagers ont aussi des téléphones intelligents et il est acceptable de proposer une interface destinée à ce type d'appareil².

¹ En fait, faut que le code client fonctionne sur mon propre poste de travail (voir la section *Comment remettre un livrable*, plus loin, pour des détails); si je dois installer le serveur localement, il doit lui aussi pouvoir fonctionner sur mon poste de travail. Vous devez, sur demande, pouvoir me donner une démonstration complète de votre SCS sur les lieux de l'université, au campus où vous suivez ce cours, et que vous puissiez donner cette démonstration peu de temps avant ou après une séance de cours avec moi.

² Je dois pouvoir tester le tout, alors sachez que j'ai un appareil Android, mais pas terriblement récent ☺

Qu'est-ce qu'un labyrinthe?

Au sens de ce projet, un labyrinthe est une surface 2D contenant des murs, des espaces vides dans lesquels une souris peut se déplacer, et des objectifs (p. ex. : des morceaux de fromage).

L'état initial d'un labyrinthe proviendra d'un média persistant (base de données, fichier XML, fichier texte, etc.) de votre choix. L'ASC s'attend à ce qu'un composant du système (un serveur de labyrinthes) soit capable d'offrir une liste des labyrinthes disponibles, de permettre de choisir l'un d'entre eux, et de livrer le contenu de celui qui aura été choisi dans un format que ses clients pourront comprendre.

Pour que l'ASC soit satisfaite, il faut que le serveur soit capable d'offrir l'accès à au moins deux labyrinthes, idéalement trois.

Qu'est-ce qu'une intelligence comportementale?

Au sens de ce projet, une intelligence comportementale est un composant capable d'interagir avec une souris et de prendre des décisions pour elle.

Pour être capable de faire ce travail, il importe donc qu'une intelligence comportementale puisse :

- Interagir avec une souris
- Obtenir de la souris l'état de son environnement (p.ex. : l'état des cases autour d'elle dans un rayon sept cases)
- Informer la souris du prochain geste à poser (p.ex. : aller au Nord, au Sud, à l'Est, à l'Ouest, rester immobile)

Pour que l'ASC soit satisfaite, il faut qu'au sens du système, chaque intelligence comportementale soit une entité externe (p.ex. : un programme distinct de celui où se déroule la simulation) et communique à « sa » souris les consignes par un canal de communication. Ceci permettra d'ajouter de nouvelles intelligences à loisir.

L'ASC s'attend à ce que vous livriez au moins deux intelligences distinctes. Ces intelligences peuvent être simples; par exemple : se déplacer de manière pseudoaléatoire, ou encore aller en ligne droite jusqu'à ce qu'on rencontre un mur, puis changer de direction de manière pseudoaléatoire. Vous pouvez bien sûr offrir une ou plusieurs intelligences plus sophistiquées si vous le souhaitez.

Qu'est-ce qu'un ensemble de règles?

Au sens de ce projet, un ensemble de règles permettra de guider le fonctionnement de la simulation. Une manière simple d'implémenter un ensemble de règles est de concevoir un serveur qui offrira des services à son client (la simulation) et lui permettra de déterminer comment le tout progresse.

Quelques exemples de règles :

- Un mouvement est accepté par souris chaque tour, par exemple chaque 500ms
- Si une souris envoie plus d'un mouvement dans un même tour, seul le plus récent d'entre eux prendra effet
- Si une souris n'envoie pas de mouvement dans un tour, alors elle est considérée immobile pour ce tour
- L'état physique d'une souris baisse d'un cran à chaque dix pas faits sans avoir mangé de fromage
- L'état physique d'une souris montre de trois crans si elle mange un fromage
- L'état psychologique d'une souris baisse d'un cran à chaque cinq pas faits sans qu'il n'y ait une autre souris dans un rayon de cinq cases
- L'état psychologique d'une souris remonte au maximum quand une autre souris se trouve dans un rayon de trois cases
- La simulation s'arrêtera après 50 tours, ou si toutes les souris ont atteint un état psychologique ou physique de zéro

Vous serez chargés de déterminer les règles implémentées dans ce composant, de même que les services que ce composant offrira.

Pour que l'ASC soit satisfaite, il faut qu'au moins deux ensembles de règles soient supportés, et que chacun permette à la simulation de suivre son cours convenablement.

Qu'est-ce qu'un déroulement de simulation?

Le cœur du système attendu est la simulation. C'est là que l'on souhaite que résident les états dynamiques de la simulation : état courant du labyrinthe, fromages restants dans le labyrinthe, état de chacune des souris dans le labyrinthe, etc.

Étant le cœur du système, l'ASC s'attend à ce que les autres composants interagissent essentiellement avec celui-ci : par exemple, le client passera par la simulation pour obtenir la liste des labyrinthes; la simulation validera l'état de la situation en transigeant avec son ensemble de règles; les intelligences artificielles serviront essentiellement de clients pour la simulation, en donnant des directives aux souris qui s'y trouvent; etc.

Pour que l'ASC soit satisfaite, il faut que ce composant (qui interagira avec plusieurs autres) soit exempt de bogues de concurrence. Pour des raisons de fluidité, l'ASC souhaite que les informations sur la simulation soient persistées (sur fichier, dans une base de données, sur un autre média à votre convenance) seulement une fois une simulation complétée.

Exemple de cas d'utilisation

Dans la vision de l'ASC, un cas d'utilisation raisonnable serait, à partir d'un client :

- Charger un labyrinthe
- Choisir deux souris
- Associer à chacune une intelligence simple
- Associer un ensemble de règles simples à la simulation
- Lancer la simulation et examiner les souris qui se déplacent et mangent du fromage, tout en constatant les variations de leur état physique et de leur état psychologique
- Une fois la simulation terminée, un rapport est présenté détaillant l'évolution de la simulation et de l'état de chaque souris dans cette simulation

Structure du travail

Vous et votre équipe jouerez le rôle d'une PME désireuse d'obtenir le (lucratif!) contrat rattaché à cet appel d'offres. Évidemment, comme tout client potentiel, ce client « voit grand ». Par conséquent, il vous faut trouver moyen à la fois d'intéresser ce client prospectif... et de livrer la marchandise promise.

Votre tâche globale sera :

- Soumettre un projet répondant pour l'essentiel aux exigences de cet appel d'offres, pour convaincre du bien-fondé de votre soumission.
- Définir deux autres livrables que vous vous engagez à rencontrer aux dates indiquées ci-dessous, le second étant le produit final que vous aurez promis initialement, et évidemment
- Livrer ce que vous aurez annoncé.

Vous aurez trois livrables ponctuels dans la session :

- Le livrable 00 décrira le détail de votre soumission. Vous définirez vous-mêmes ce en quoi consisteront chacun des autres livrables.
- Le livrable 01 devra faire la démonstration que votre projet progresse de manière raisonnable, et
- Le livrable 02, qui sera votre produit fini. Entendu que ce qui est attendu est un prototype utilisable comprenant ce que vous avez annoncés.

On parle donc :

- D'un livrable au début du cours S03.
- D'un livrable au début du cours S08, et
- D'un livrable au début du cours S12.

Composition des équipes

Les équipes de travail pourront comprendre entre 1 et 4 personnes inclusivement; j'aime bien qu'il y ait toujours au moins deux équipes par classe, pour qu'il soit possible d'échanger en fin de parcours sur les choix qui ont été faits. Votre soumission doit, pour être recevable, rencontrer l'essentiel des demandes formulées dans l'appel d'offres (vous pouvez en exclure certaines parties, dans la mesure où ces exclusions sont expliquées et justifiées, et dans la mesure où le travail de base – voir ci-dessous – est fait), et doit expliquer votre vision du système, de ses composants, et de la manière dont ces composants communiqueront.

Ce travail vise un double objectif, soit celui :

- De partager les responsabilités techniques, architecturales et de gestion avec des pairs, et
- De partager les fruits de vos efforts avec vos pairs en fin de parcours (on souhaite donc ardemment qu'il y ait au moins deux équipes!)

Une soumission, pour être recevable, doit aussi permettre au travail de base d'être fait. **Le travail de base est la production système permettant de simuler l'évolution de l'état de souris, quand chacune est guidée à travers un labyrinthe par une intelligence artificielle, tout en visualisant à la fois l'évolution de la simulation et, ultérieurement, ses résultats.**

Une soumission non recevable sera refusée et donc considérée en retard pour ce qui est des règles de comptabilisation du pointage.

Gestion et suivi du travail d'équipe

Pour assurer que le travail d'équipe se fasse, vous devrez :

- Tenir des minutes de chacune de vos rencontres. Ces minutes devront être datées et signées par tous les membres de l'équipe. Dès la première rencontre, assurez-vous de mettre au point des règles qui vous permettront de résoudre les conflits s'il en survient. Vous devrez pouvoir déposer les minutes à jour sur demande.
- Identifier l'auteur ou les auteurs dans chaque document, chaque fichier de code et dans toute autre production pertinente de votre équipe.
- Tenir à jour un registre des documents, fichiers de code et autres productions pertinentes de même que de leurs auteurs³, qui pourra en tout temps être exigé par le professeur.
- Le professeur s'attend à ce que toutes et tous contribuent à la fois à l'aspect administratif, à l'aspect humain et à l'aspect technique du projet, dans les limites des capacités de chacune et de chacun de même que dans le respect des objectifs du cours. Si un membre d'une équipe ne fait que dessiner des icônes ou ne contribue en rien au volet CS du projet, alors le professeur pourra, à sa discrétion, ajuster la note de cet individu.

Dans l'éventualité où la composition de l'équipe changerait (démission, mésentente, des trucs moches mais qui arrivent), je compte sur l'intégrité de tous les gens impliqués pour que le travail fait par chaque individu reste identifié comme tel. Je traiterai ces incidents au cas par cas, mais ils devraient être rares (dans un monde idéal, il n'y en aurait aucun).

³ Un tableau 2D dans *Excel* peut très bien faire l'affaire, mais vous pouvez être plus sophistiqués si vous en avez envie ou si vous en ressentez le besoin.

Comptabilité et évaluation du projet

La note accordée pour ce projet tiendra compte d'un certain nombre de facteurs :

- Respect des échéances.
- Progression acceptable du projet.
- Qualité des propriétés du projet telles que livrées.
- Qualité du travail individuel.
- Qualité du travail d'équipe.

Contraintes techniques particulières

Le produit complet, pour être jugé recevable, devra intégrer au moins deux langages de programmation distincts⁴ et au moins une banque de données.

Il devra permettre aux chercheurs des universités de collaborer à travers l'outil sans avoir à se déplacer physiquement.

Pour le livrable 00, au début de la séance S03, chaque équipe devra déposer :

- Les noms, les courriels⁵ et les responsabilités⁶ des membres de l'équipe, respectant les contraintes ci-dessus. Identifiez clairement une personne contact avec qui je transigerai et qui relayera les messages au reste de l'équipe.
- Une analyse du système à réaliser.
- Une proposition d'un prototype opérationnel qui pourra être livré au début de la séance S12, respectant les contraintes ci-dessus, et
- Le détail du livrable 01, dû au début de la séance S08.

Il est entendu que le livrable 02, dû au début de la séance S12, sera constitué du produit complet promis par votre équipe, et devra être opérationnel. Ce livrable devra être accompagné d'un guide d'installation et d'un guide d'utilisateur (les deux peuvent ne faire qu'un). Ces guides font partie du livrable et en influenceront la note.

Si le tout est livré à temps et accepté par le client prospectif⁷, votre équipe recevra **20** points. Si le tout est livré en retard ou est jugé irrecevable (et donc déposé en retard), alors votre équipe recevra **10** points. Des ajustements mineurs pourront être demandés par le client prospectif sans que cela n'entraîne de pénalité pour votre équipe.

Livrer une analyse et une soumission recevables constitue une condition *sine qua non* à la poursuite du projet (donc à la collecte des points subséquents).

Les livrables 01 vaudra **30** points si livré à temps et dans le respect des règles⁸. Une livraison en retard ou ne respectant pas les règles vaudra **15** points et pourra être retournée à l'équipe pour ajustements. Le livrable 02 vaudra **50** points, ou **25** points si livré en retard, inacceptable ou devant être retourné à l'équipe.

Pour chacun des livrables, le professeur pourra accorder un ajustement de $\pm 20\%$ en fonction de la qualité du travail.

⁴ Non, deux langages .NET ne sont pas suffisamment distincts pour qualifier ici. Et non, HTML n'est pas un langage de programmation. Cela ne signifie pas que vous ne puissiez pas explorer HTML5, par exemple, mais il importe que le défi technique de programmation et d'intégration demeure. Dans le doute, consultez l'enseignant.

⁵ Entendu ici que le courriel donné doit être opérationnel et doit être celui que je pourrai utiliser pour vous rejoindre.

⁶ Entendu que tous doivent être impliqués sur le plan technique comme sur les autres plans pour que le projet soit considéré recevable.

⁷ Simulé gracieusement par votre chic professeur.

⁸ Entendu ici l'union des règles de l'activité pédagogique en tant que telle et les règles que vous aurez vous-mêmes indiquées dans votre livrable 00.

Ce que les anglophones nomment un Reality Check

Malgré tout votre talent, tout votre bon vouloir, et malgré le fait que vous avez droit à un chic professeur, il reste que le travail à faire est difficile, que ce cours est demandant, que le temps passe et que vous avez des obligations extérieures à ce cours⁹.

Il est donc réaliste de présumer qu'à un moment ou l'autre, des irritants placés sur votre route vous empêcheront de livrer exactement ce que vous aviez promis. Idéalement, votre planification sera prudente et réaliste, mais évaluer le temps véritablement requis pour réaliser une tâche est une chose qui se fait plus avec l'expérience qu'avec des cours.

Nous avons tous tendance à nous percevoir de manière romantique, que ce soit en surévaluant ou en sous-évaluant notre capacité à livrer. De plus, vous devez évaluer le temps qu'il vous faudra pour réaliser un SCS avant d'avoir suivi le cours portant sur le sujet! Mais nous n'avons pas le choix d'y aller ainsi si nous voulons retirer du cours des apprentissages pertinents. Il nous faudra nous lancer dans l'action, et viser à retirer un maximum de l'expérience.

Cela dit, une des particularités d'un diplôme comme celui où vous êtes inscrit(e)s est qu'une livraison en retard y est inacceptable : un retard signifie « pas de note ».

Comment gérons-nous cela? Avec souplesse, évidemment. Ainsi, vous aurez le droit de diverger *légèrement* de votre route¹⁰ telle que tracée *par vous* au livrable 00. Vous devrez, par contre, justifier ces divergences par un document (un par équipe et par livrable, pas plus).

Un bon truc pour que ce ne soit trop lourd à faire est de tenir à jour un échéancier de projet¹¹, et d'avoir une rencontre hebdomadaire¹² pour suivre la progression de tous et chacun. Ainsi, vous pourrez sauvegarder les échanges et il ne restera plus, au besoin, qu'à mettre le tout en forme pour livrer un résumé de la situation.

Je vous suggère de faire le suivi du projet de manière rotative. Échangez-vous le rôle de *gestionnaire d'office pour le projet* sur une base hebdomadaire, en prenant toujours soin de garder plus d'une copie de sauvegarde des documents au cas où un accident surviendrait. Si vous n'êtes pas habitué(e)s aux outils de gestion de projet, un truc serait d'y aller avec des *cogestionnaires* à chaque semaine, pour que personne ne soit pris au dépourvu et que le suivi soit toujours bien fait.

Je me réserve le droit d'être tolérant pour des divergences justifiées formellement et raisonnables. Ça fait partie de la vie. Je présume une bonne foi de la part de tous les intervenants. Si j'ai l'impression que ça dérape, que des gens ne contribuent pas à leur équipe ou, à l'inverse, que des gens ne permettent pas à leurs coéquipiers de s'impliquer, je me réserve le droit d'intervenir sur le plan humain comme sur le plan des notes. Reste que la vie est beaucoup plus agréable quand tout le monde s'implique.

⁹ ...que ce soit un emploi, une famille, une passion pour le bingo, peu importe.

¹⁰ ...dans la mesure où vous livrez un système correspondant au moins aux exigences du travail de base, plus haut.

¹¹ ...par exemple avec *Microsoft Project*, mais il existe des outils à code ouvert pour faire le même travail.

¹² ...quitte à ce qu'elle soit électronique.

Comment remettre un livrable

Les livrables peuvent être livrés sous forme imprimée ou par courriel à mon adresse @USherbrooke.ca, et il en va de même pour les documents justifiant les divergences avec le projet tel que soumis. Vous pourrez, pour obtenir des commentaires, me livrer du code sous forme imprimée, mais vous ferez cela à votre convenance (à moins que je ne vous en fasse expressément la demande). Ne me submergez pas, je vous prie: essayez de livrer ce pour quoi vous aimeriez recevoir du feedback, pas la totalité du code produit par votre équipe. Je donnerai typiquement une rétroaction électronique ou papier en fonction du format de la remise.

Si vous souhaitez une rétroaction plus riche pour un livrable électronique, privilégiez un document en format `.docx` ou `.pdf`, mais notez que les annotations sont typiquement plus faciles à suivre dans un `.docx` que dans un `.pdf`.

Je m'attends à recevoir sous forme électronique des livrables que je peux tester chez moi¹³. Si je dois accéder à des serveurs externes, vous devez faire en sorte que ce soit agréable et fluide. Vous devez aussi vous assurer, par souci d'intégrité, de pouvoir me produire les sources de tous vos modules sur demande et de pouvoir me donner une démonstration sur place, au campus où vous suivez ce cours et un soir où je suis présent, si j'en exprime le souhait; si je le fais, je vous donnerai quelques jours d'avis.

Ne me rendez pas la vie trop compliquée je vous prie : la correction de tout ceci sera un contrat en soi. Il est nécessaire que je puisse installer et désinstaller le tout aussi simplement que possible, que ce soit par un installateur automatisé ou à l'aide de consignes simples et claires. Je corrige énormément de travaux chaque session; il est donc impératif que vous me rendiez la tâche facile à l'installation et que mon ordinateur soit *propre* après la désinstallation.

Tout livrable devra présenter un français de qualité. Une police de caractères non proportionnelle (quelque chose comme Courier New ou Lucida Console par exemple) devra être utilisée pour le code source. Ces polices conviennent bien à un tel usage, clarifiant l'indentation appliquée. D'ailleurs, même si vous utilisez des outils de génération automatique de code, ce code devra être indenté de manière lisible, sinon il vous sera retourné sans que je ne l'aie lu.

Le facteur « surprise »

Il se peut (!) que s'ajoutent en chemin des consignes particulières, comme par exemple l'intégration d'un module produit par un tiers (simulé par votre chic professeur), le tout à la demande du client prospectif. Cela fait partie de la vie, et fait aussi partie de l'énoncé de l'activité pédagogique. Vous voilà avertis!

Amusez-vous follement!

¹³ PC au processeur Intel 64 bits à huit cœurs cadencés à 2,7 GHz, 16 Go RAM, Microsoft Windows 10. Notez que, en particulier pour l'accès à une base de données, il doit être possible pour moi d'utiliser votre système de la maison, alors veillez à prendre des dispositions en conséquence.