**Soumission de projet intégrateur**

***La science des fusées pour les nuls***

*par*

Thomas Corbeil et James Huynh

Équipe 25

*Travail présenté à*

Caroline Houle

420-SCD Intégration des apprentissages en SIM

Groupe 2

Collège de Maisonneuve

8 février 2017

**1. Introduction**

Ce document de soumission visera à répondre aux questions entourant tous les aspects à propos de la conception et correction de notre application nommée *La science des fusées pour les nuls.* Premièrement, nous présenterons brièvement l’application. Deuxièmement, nous examinerons l’aspect scientifique lié au développement de ce projet. Troisièmement, nous explorerons les diverses fonctionnalités techniques de l’application. Quatrièmement, nous présenterons un diagramme UML représentant le code prévu de l’application. Cinquièmement, nous présenterons un échéancier concernant les diverses phases de développement du programme. Pour terminer, nous examinerons les fonctionnalités optionnelles du programme qui pourront être codées en cas de surplus de temps.

**2. Présentation de l’application**

**2.1 Nature du logiciel et contexte d’utilisation**

L’application permettra de simuler le décollage et le vol orbital d’une fusée de manière simplifiée et permettra aussi de construire une fusée à partir d’un ensemble de pièces défini préalablement au décollage de la fusée. Dû à la simplification de divers concepts de physique mécanique, le programme sera plutôt un jeu éducatif visant à instruire les utilisateurs à propos de certains concepts de physique mécanique s’appliquant aux fusées. L’application sera principalement destinée aux amateurs de vols spatiaux.

**2.2 Recherche documentaire**

Nous en sommes venus à cette idée en jouant à divers jeux de vol spatial réaliste. Nous avons étés principalement inspirés par un simulateur nommé *Kerbal Space Program* et un jeu sur tablette nommé *Space Agency.* Nous avons aussi été inspirés par diverses chaînes éducatives sur Youtube traitant du sujet. Nous avons utilisé les diverses ressources du cours de physique mécanique donné au collège de Maisonneuve.

**3. Teneur scientifique du projet**

**3.1 Matière utilisée**

Notre application touche aux disciplines de la physique mécanique, afin de calculer correctement la trajectoire de la fusée, et de l’informatique, pour construire la fusée à partir de différentes pièces. Premièrement, nous utiliserons le principe de superposition des forces pendant tout le vol de la fusée afin de calculer la force résultante sur la fusée. Deuxièmement, nous utiliserons la première, la seconde et la troisième loi de Newton pendant tout le vol afin d’obtenir l’accélération et la direction de la fusée. Finalement nous utiliserons aussi les lois de la cinématique afin de trouver la vitesse.

Seconde loi de newton:

*∑F = ma*

Équations de la cinématique:

*v = v0 + at*

*x = x0 + v0t + ½at2*

*v2 = v02 + 2a(x - x0)*

**3.2 Modèles, limites et autres simplifications**

Premièrement, nous négligerons la force de friction sur la fusée. Nous ferons aussi en sorte que la force exercée par le moteur de la fusée soit constante pendant toute la durée du vol. Aussi, la simulation sera limitée à deux dimensions soit les plan x et y pour le décollage (vue de côté) et y et z pour le vol orbital (vue de dessus). Concernant l’orbite elle-même, pour simplifier son implémentation, la trajectoire de l’orbite suivra un cercle au lieu d’une ellipse. Cela veut dire que les Lois de Kepler ne s’appliqueront pas (Lois de l’espace) et seront substituées par la mécanique de base. De plus, pour maintenir une cohérence et du réalisme dans l’application, tout déplacement (x et z) durant le vol orbital sera compensé par une force «automatique» sur le plan complémentaire, par exemple, comme lorsque l’utilisateur veut rapprocher sa fusée vers la Terre pendant le vol orbital. Par conséquent, pour maintenir la trajectoire circulaire, la vitesse tangente à la trajectoire sera augmentée. En d’autres mots, il y aura une force qui va accélérer la fusée momentanément.

**3.3 Complexité informatique**

La majorité de la complexité informatique de notre projet repose dans l’éditeur de fusée. En effet, l’utilisateur pourra assembler sa fusée de façon personnalisée en transportant les pièces qu’il désire ajouter à sa fusée à partir d’un ensemble de pièces jusqu’à la zone d’assemblage. Le choix de pièces n’affectera pas seulement l’aspect esthétique de la fusée mais aussi ses performances et autres variables telles que la masse, la quantité de carburant et la poussée générée par la fusée. Aussi, s’il manque une pièce critique à la fusée, l’éditeur devra empêcher l’utilisateur de procéder à la prochaine étape. De plus, une partie de la complexité résidera dans la gestion des diverses images utilisées pour dessiner la fusée et dans l’optimisation du chargement des images. Finalement, le reste de la complexité informatique résidera surtout dans l’animation du vol de la fusée et dans le calcul des diverses variables de la fusée qui sont dépendantes des pièces choisies mais aussi dans l’ajout de sons pertinents à la situation.

**4. Fonctionnalités de l’application**

**4.1 Fonctionnement général**

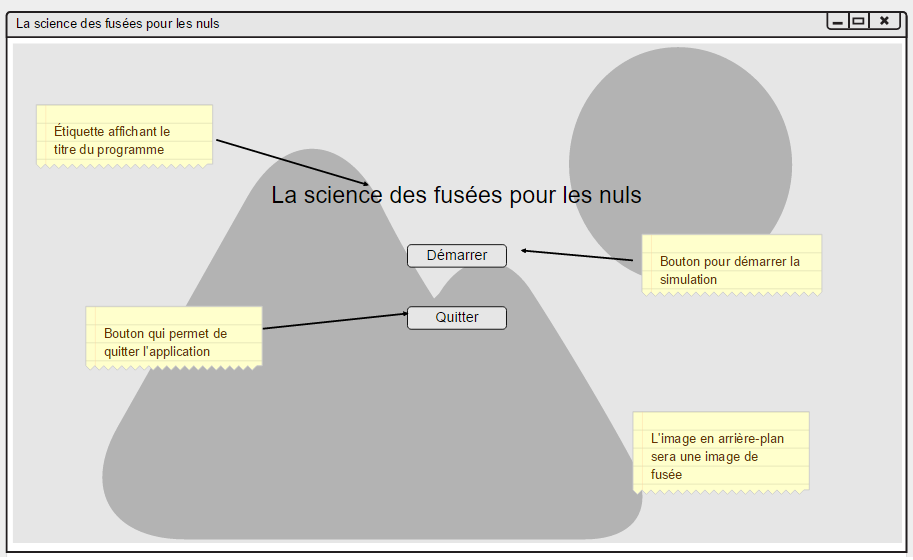
L’utilisateur ouvrira l’application pour se rendre sur le menu et appuiera sur “commencer” pour démarrer la simulation. Il se retrouvera sur l’écran de construction de la fusée où il pourra la personnaliser à son goût et voir ses caractéristiques. S’il manque des pièces, l’utilisateur ne pourra pas accéder à la prochaine phase, sinon il continuera vers le décollage. À la phase de décollage, l’utilisateur pourra appuyer sur le bouton «décollage». Un décompte de dix secondes suivra. Ensuite, la fusée décollera et l’utilisateur pourra ajuster divers paramètres concernant le vol de la fusée. S’il atteint l’altitude et la vitesse horizontale requises, le vol passera à la phase orbitale. Durant le vol orbital, l’utilisateur pourra ajuster sa trajectoire en contrôlant un *système de contrôle par réaction* (RCS en anglais :<https://fr.wikipedia.org/wiki/Reaction_Control_System>). Ce système permettra de faire accélérer la fusée dans les directions parallèles aux axes y et z. Dernièrement, à partir de la phase orbitale, l’utilisateur pourra retourner à la première phase de l’application, qui est la phase de construction, en cliquant sur un bouton prévu à cet effet. Dans l’éventualité où l’utilisateur échoue le décollage ou le vol orbital, la simulation se terminera obligeant l’utilisateur à recommencer.

**4.2 Entrées et sorties**

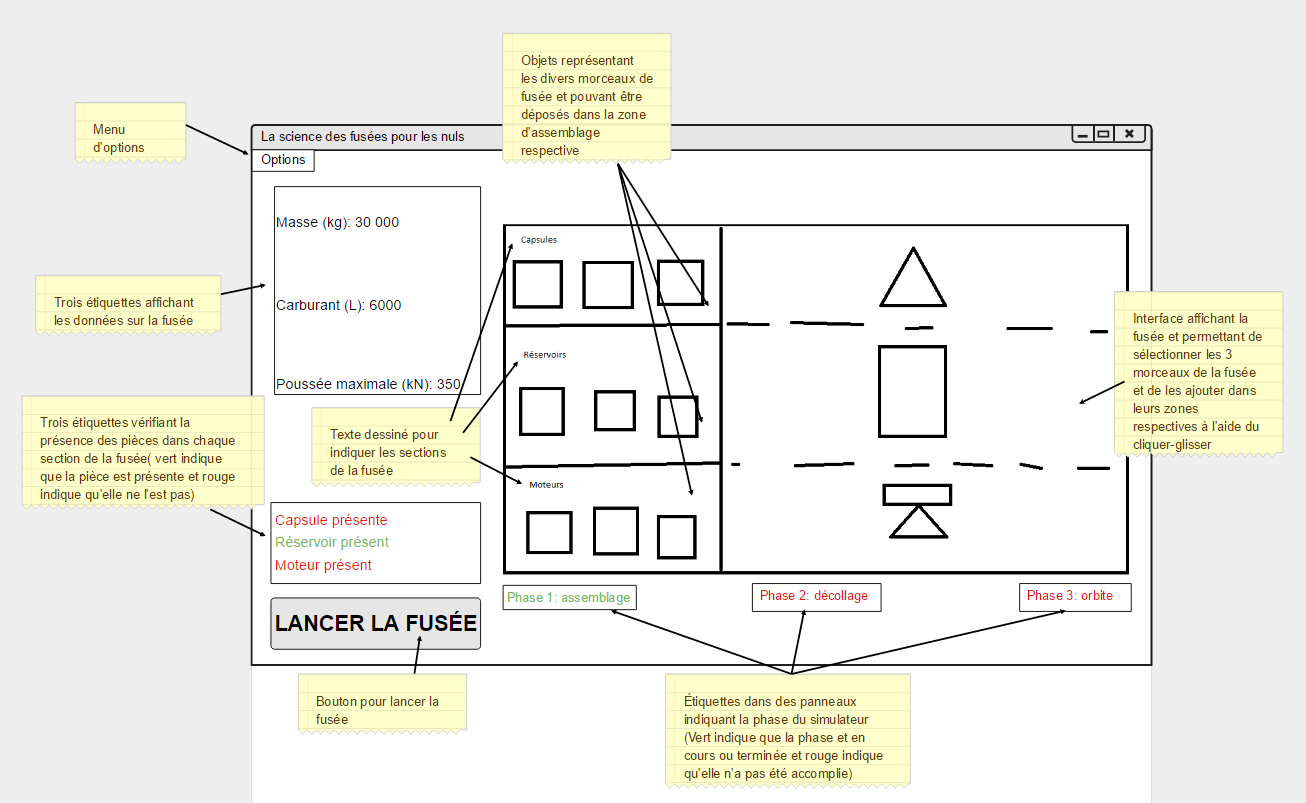
|  |  |
| --- | --- |
| Entrées | Sorties |
| * Choix de la capsule de la fusée (section supérieure) * Choix du réservoir de la fusée (section du milieu) * Choix du moteur de la fusée (section inférieure) * Pourcentage de poussée * Inclinaison de la fusée pendant le décollage * Direction de la fusée pendant l’orbite | * Masse (en kg) de la fusée pendant l’assemblage * Quantité de carburant (en L) de la fusée pendant l’assemblage * Poussée maximale (en kN) de la fusée pendant l’assemblage. * Présence des trois éléments critiques (capsule, réservoir et moteur) pendant l’assemblage * Phase actuelle du simulateur * Apparence de la fusée en fonction des diverses pièces choisies pendant l’assemblage * Pourcentage de poussée de la fusée pendant le décollage * Pourcentage de carburant restant pendant le décollage * Accélération en X (en m/s^2) pendant le décollage * Accélération en Y (en m/s^2) pendant le décollage * Altitude de la fusée (en m) pendant le décollage * Poussée en X (en kN) pendant le décollage * Poussée en Y (en kN) pendant le décollage * Masse (en kg) de la fusée pendant le décollage * Vitesse en X (en m/s) pendant le décollage * Vitesse en Y (en m/s) pendant le décollage * Visualisation de l’angle et de l’altitude de la fusée (à l’aide d’une animation) pendant le décollage * Pourcentage de poussée de la fusée pendant le vol orbital * Pourcentage de carburant restant pendant le vol orbital * Accélération latérale (en m/s^2) pendant le vol orbital * Accélération horizontale (en m/s^2) pendant le vol orbital * Altitude de la fusée (en m) pendant le vol orbital * Masse (en kg) de la fusée pendant le décollage * Vitesse latérale (en m/s) pendant le vol orbital * Vitesse verticale(en m/s) pendant le vol orbital * Visualisation de l’orbite et des changements de trajectoire de la fusée par l’utilisateur à l’aide d’une zone de dessin * Texte indiquant la réussite ou l’échec du vol * Sons de l’application |

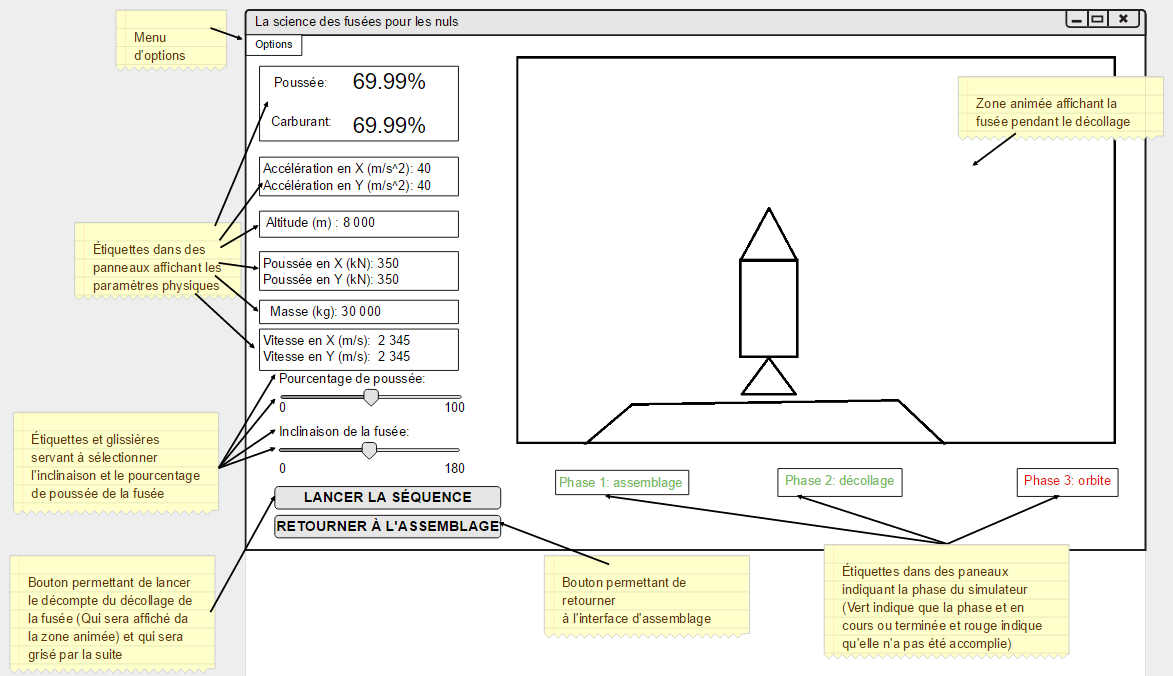
**4.3 Maquettes**

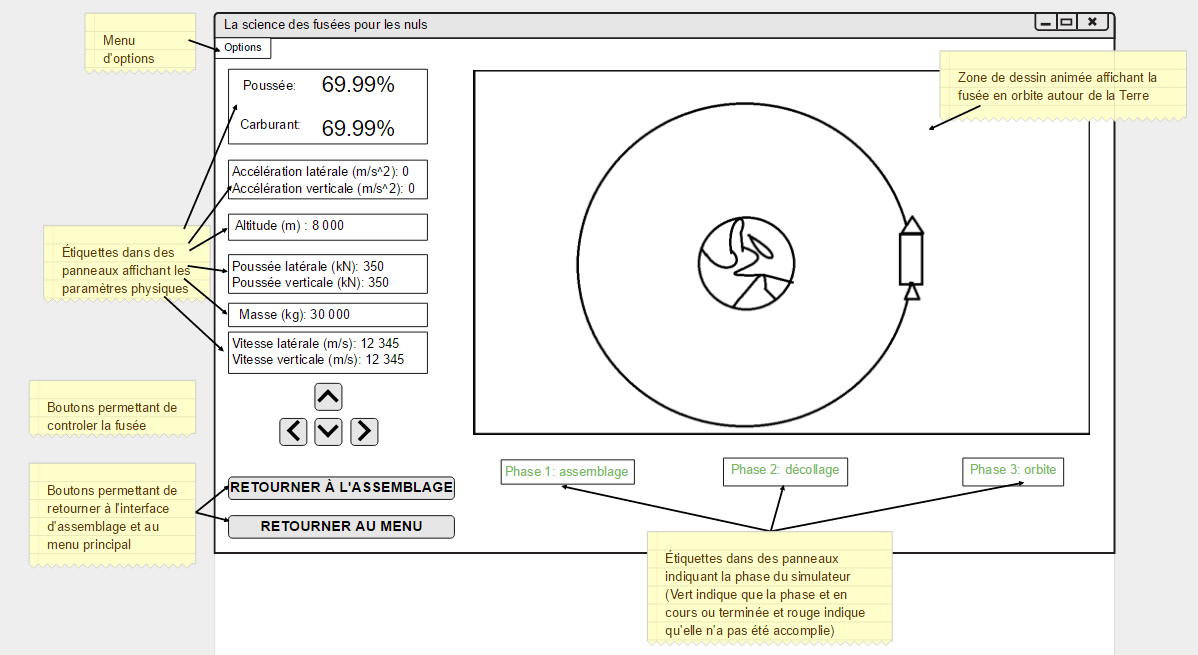
Voici l'ensemble des maquettes présentées pour cette application :

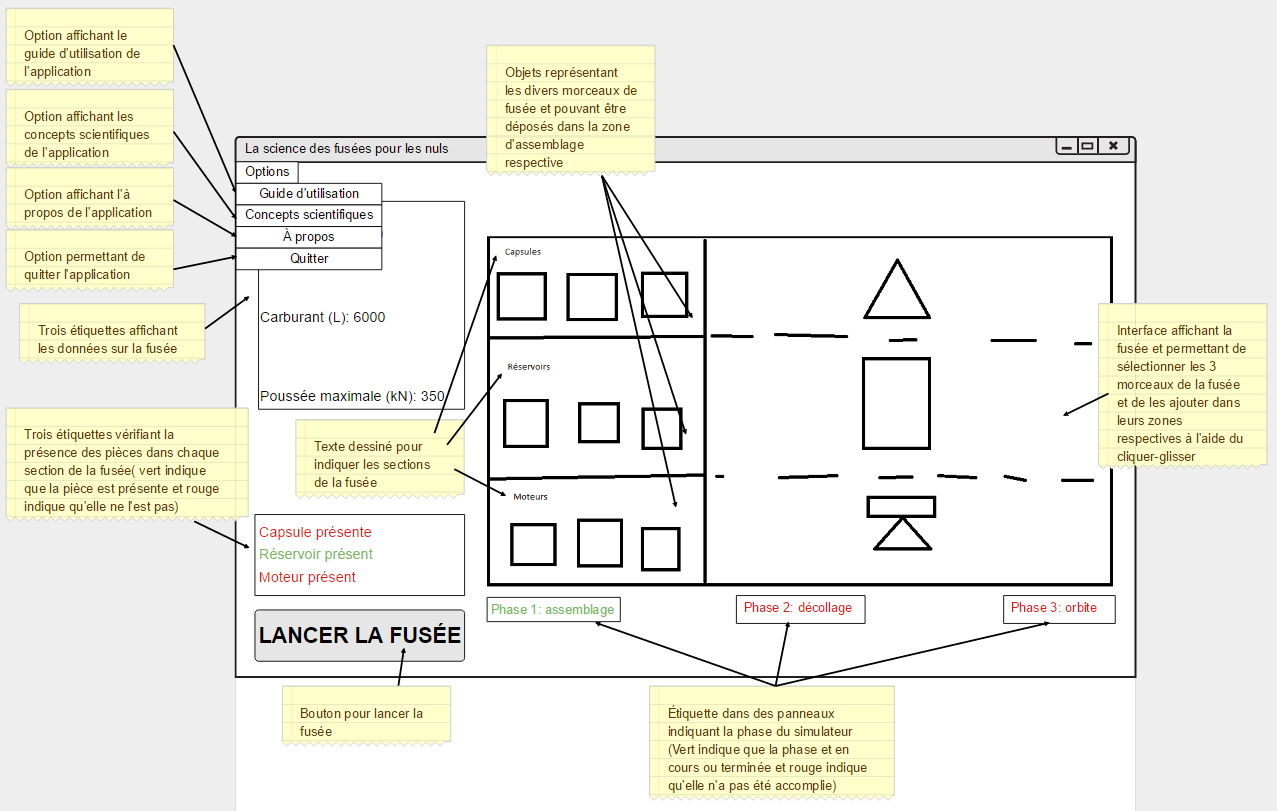
Maquette M1 : Menu principal

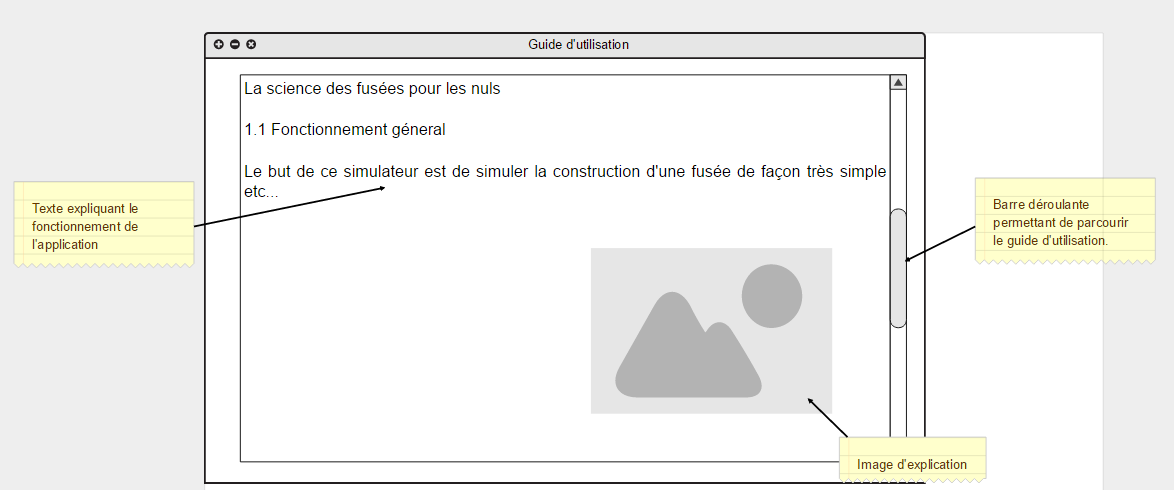
Maquette M2 : Phase de construction

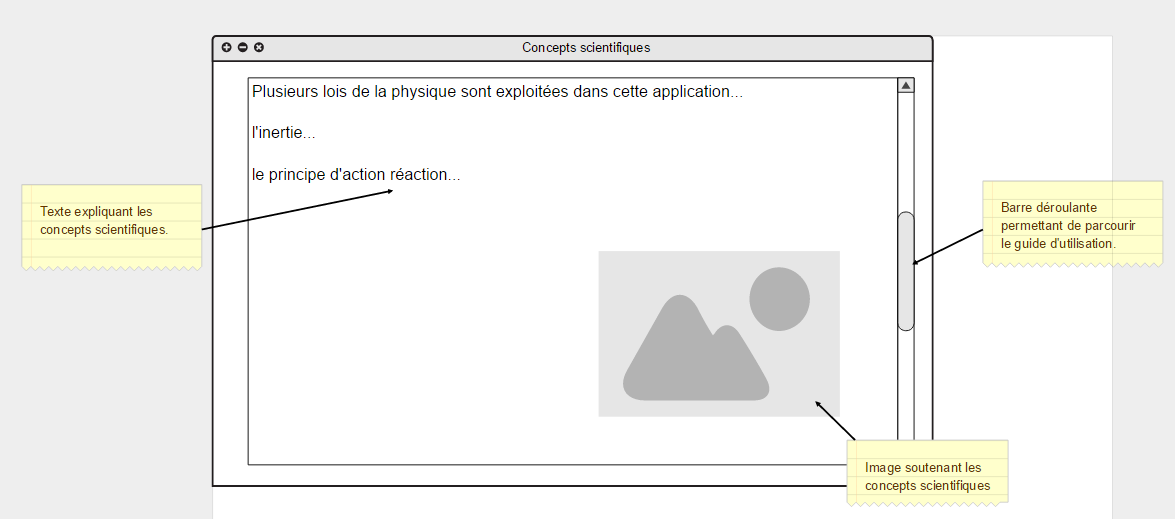


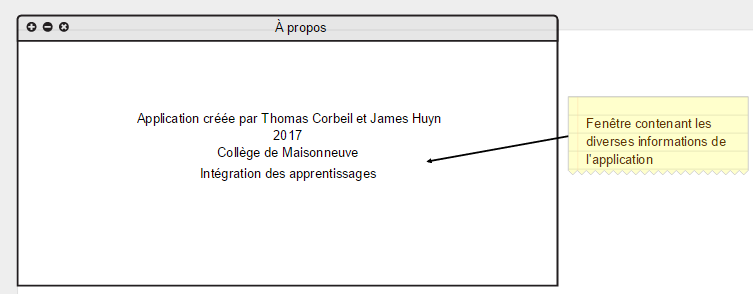
Maquette M3 : Phase de décollage

Maquette M4 : Phase de vol orbital

Maquette M5 : Aperçu du menu d’options

Maquette M6 : Menu du guide d’utilisation

Maquette M7 : Menu des concepts scientifiques

Maquette M8 : Menu à propos

**4.4 Scénarios d’utilisation**

L'application permet les différentes fonctionnalités suivantes:

S1: Démarrer le simulateur

S2: Construire la fusée

S3: Lancer la fusée

S4: Échouer le lancement

S5: Naviguer dans l’espace

S6: Retourner à l’assemblage

S7: Détruire la fusée

S8: Quitter l’application

S9: Retourner au menu

S10: Sélectionner un menu d’aide

Scénario S1: Démarrer le simulateur

Pré-condition(s):

* L’utilisateur utilise la fenêtre M1

Échanges:

* L’utilisateur appui sur le bouton «Démarrer» de la fenêtre M1
* L’application passe alors à la fenêtre M2
  + S’il le souhaite, l’utilisateur peut aussi quitter l'application à l’aide du bouton quitter de la fenêtre et se retrouve dans le scénario S8

Post-condition(s):

* L’utilisateur se retrouve sur la fenêtre M2

Scénario S2: Construire la fusée

Pré-condition(s):

* L’utilisateur utilise la fenêtre M2

Échanges:

* L’utilisateur construit sa fusée en prenant les pièces qu’il désire de la liste dans la zone de dessin de la fenêtre M2 et en les déposant (grâce au cliquer-glisser) dans la zone de construction
  + Il doit choisir une pièce de chacune des trois catégories sans quoi l’application l’empêchera de poursuivre à la phase suivante
  + S’il désire changer une des pièces choisies auparavant, il peut simplement la prendre de la zone de construction et la remettre dans la liste (grâce au cliquer-glisser) pour la retirer
* Au fur et à mesure que l’utilisateur choisit les pièces, les statistiques de la fusée et les indicateurs indiquant si les pièces sont présentes ou non sont mis à jour.
* Lorsque l’utilisateur est prêt à lancer sa fusée, il appuie sur le bouton «Lancer la fusée» de la fenêtre M2
  + Le bouton sera grisé s’il manque une ou plusieurs pièces à la fusée et ce, jusqu’à ce qu’il sélectionne les pièces manquantes et les dépose ( à l’aide du cliquer-glisser) dans la zone de construction
* Après avoir appuyé sur le bouton lancer la fusée, le texte «Phase 2: décollage» devient vert et il y a une transition dans la zone de dessin jusqu’à la prochaine phase

Post-condition(s):

* L’utilisateur se retrouve sur la fenêtre M3

Scénario S3: Lancer la fusée

Pré-condition(s):

* L’utilisateur utilise la fenêtre M3

Échanges:

* L’utilisateur fait décoller la fusée en cliquant sur le bouton «Lancer la séquence»
  + Un décompte de dix secondes est alors affiché dans la zone de dessin
  + Il peut ajuster l’angle de la fusée pendant tout le vol à l’aide de la glissière «Inclinaison de la fusée» et des touches directionnelles gauche et droite
  + Il peut ajuster le pourcentage de poussée de la fusée pendant tout le vol à l’aide de la glissière «Pourcentage de poussée» ou des touches directionnelles haut et bas
* Tout au long du vol, le pourcentage de poussée, le pourcentage de carburant, l’accélération de la fusée, l’altitude, la poussée, la masse et la vitesse de la fusée seront mis à jour
  + Plus l’altitude de la fusée est élevée, plus l’arrière-plan deviendra foncé
* Si l’utilisateur réussi à atteindre la vitesse en X et l’altitude prescrite, le vol passe à la phase orbitale grâce à une transition dans la zone de dessin et le texte «Phase 3: vol orbital» devient vert.
  + Dans le cas contraire, l'utilisateur se retrouve dans le scénario S4

Post-condition(s):

* L’utilisateur se retrouve sur la fenêtre M4

Scénario S4: Échouer le lancement

Pré-condition(s):

* L’utilisateur utilise la fenêtre M3
* La fusée de l’utilisateur est en vol

Échanges:

* L’utilisateur échoue le lancement en ne parvenant pas à satisfaire les conditions de vitesse en X et d’altitude prescrite ou en manquant de carburant avant d’atteindre l’altitude prescrite
* Un message s’affiche dans la zone de dessin indiquant que la simulation est un échec et le texte Phase 2 décollage devient rouge
  + L’utilisateur peut retourner à l’étape d’assemblage en appuyant sur le bouton «Retourner à l’assemblage» et se retrouve dans le scénario S6

Post-condition(s):

* Aucune

Scénario S5: Naviguer dans l’espace

Pré-condition(s):

* L’utilisateur utilise la fenêtre M4

Échanges:

* L’utilisateur déplace (axe Z et Y ) la fusée en cliquant sur les boutons directionnels ou en appuyant sur les flèches du clavier. Cela modifie la position verticale de la fusée et sa vitesse horizontale
  + Si la fusée se retrouve en-dessous de l’altitude de l’atmosphère, l’utilisateur se retrouve au scénario S7

Post-condition(s):

* Aucune

Scénario S6: Retourner à l’assemblage

Pré-condition(s):

* L’utilisateur utilise la fenêtre M3 ou M4

Échanges:

* L’utilisateur choisit de retourner à l’assemblage en cliquant sur le bouton approprié
* Le texte pour toute les phases autre que la «Phase 1: Assemblage» devient rouge

Post-condition(s):

* L’utilisateur retourne au scénario S2

Scénario S7: Détruire la fusée

Pré-condition(s):

* L’utilisateur utilise la fenêtre M3 et rentre dans l’atmosphère

Échanges:

* Une animation de la fusée en train d’exploser se déroule dans la zone de dessin de la fenêtre M4
* Le texte Phase 3: orbite devient rouge
  + S’il le veut, l'utilisateur peut appuyer sur le bouton «Retourner à l’assemblage» et se retrouve dans le scénario S6

Post-condition(s):

* Aucune

Scénario S8: Quitter l’application

Pré-condition(s):

* L’utilisateur utilise la fenêtre M1, M2, M3 ou M4

Échanges:

* L’utilisateur clique sur le bouton «quitter» de la fenêtre M1 ou sélectionne l’option «Quitter» dans le menu «Options» des fenêtres M2, M3 ou M4
* L'application se ferme

Post-condition(s):

* Aucune

Scénario S9: Retourner au menu

Pré-condition(s):

* L’utilisateur utilise la fenêtre M4

Échanges:

* L’utilisateur clique sur le bouton «retourner au menu»

Post-condition(s):

* L’utilisateur se retrouve à la fenêtre M1

Scénario S10: Sélectionner un menu d’aide

Pré-condition(s):

* L’utilisateur utilise la fenêtre M2, M3 ou M4

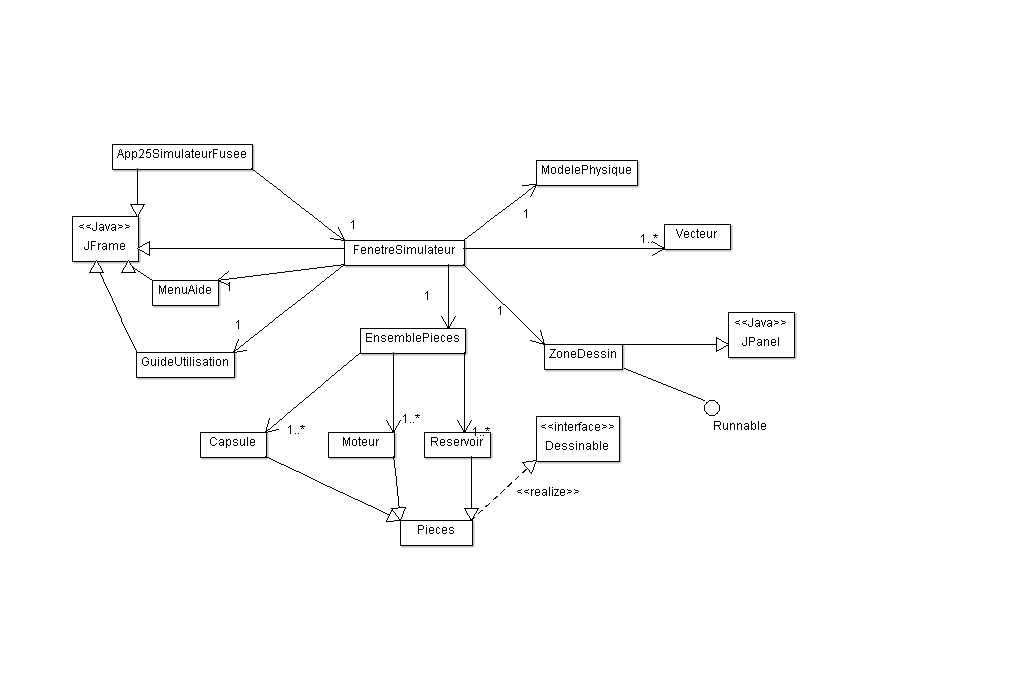
Échanges:

* L’utilisateur sélectionne une option autre que l’option quitter dans le menu «options»
  + Si l’utilisateur sélectionne l’option «Guide d’utilisation», une fenêtre contenant le guide d’utilisation apparaîtra par dessus la fenêtre principale.
  + Si l’utilisateur sélectionne l’option «Concepts Scientifiques», une fenêtre contenant les concepts scientifiques apparaîtra par dessus la fenêtre principale.
  + Si l’utilisateur sélectionne l’option «À propos», une fenêtre contenant les informations à propos de l’application apparaîtra par dessus la fenêtre principale.

Post-condition(s):

* Une fenêtre secondaire appropriée apparaîtra par dessus la fenêtre principale

**5. Diagramme de classes UML préliminaire**



**6. Échéancier**

|  |  |  |
| --- | --- | --- |
| Semaines  de la session | Nouvelles fonctionnalités implantées  et démontrables | Remarques/étape-clé |
| 5 –6 | Le module de construction est fonctionnel mais contient un ensemble de pièces limité.  La fusée peut seulement voler en ligne droite. L’orbite n’est pas modifiable. | Premier démo concret d'un cas simplifié (faisabilité) |
| 7 | La phase de vol est complètement fonctionnelle (on peut orienter la fusée et modifier la poussée du moteur). | Remise de la version *pré-alpha* |
| 9 | La phase de vol orbital est complètement fonctionnelle. Tous les composantes affichant les informations de la fusée en vol sont fonctionnelles. | Remise de la version *alpha* |
| 11 | Le module de construction est terminé et fonctionnel et les guides d’aide contiennent du texte. Les dessins de la fusée sont terminés. | Remise de la version *beta* |
| début 13 | Tous les bugs sont résolus. L’esthétisme de l’application aura été finalisé. L’application est complètement blindée. | Remise finale du projet |

**7. Fonctionnalités optionnelles**

La première fonctionnalité optionnelle que nous voudrions implémenter dans notre application serait un système permettant la sauvegarde du design d’une fusée afin de pouvoir la charger dans l’application plus tard. La seconde fonctionnalité optionnelle que nous aimerions implémenter serait un système permettant d’alterner entre une vue explosée et une vue normale de la fusée pendant la phase de construction. Finalement la dernière fonctionnalité que nous aimerions implémenter serait une phase de réentrée de la fusée dans l’atmosphère.

**8. Conclusion**

En somme, ce projet présente un défi de taille pour nous en raison des diverses techniques de programmation inexplorées auparavant, comme la gestion d’une base de donnée d’images. Dans les semaines à venir, nous aurons beaucoup de travail à effectuer; la structure du projet est primordiale à établir. Il faudra aussi arriver à surmonter des problèmes en trouvant des solutions originales. Malgré ce périple difficile, cette aventure informatique fera de nous, jeunes idéalistes, de meilleurs programmeurs avec plus d’expérience de vie.

**9. Grille de correction**

**Grille de correction: Soumission du projet** Numéro d'équipe: 25

*Critères utilisés pour la correction:*

ü *Facilité pour le lecteur à se faire une idée claire et précise du fonctionnement*

ü *Richesse, pertinence, et clarté des propos*

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Élément de correction** | **Pondération** | **Pts obtenus** |
| 1 | Introduction au document | 3 |  |
| 2 | Présentation de l'application: |  |  |
| 2.1 | · Nature du logiciel et contexte d'utilisation | 7 |  |
| 2.2 | · Recherche documentaire | 5 |  |
| 3 | Teneur scientifique |  |  |
| 3.1 | · Matière utilisée | 6 |  |
| 3.2 | · Modèles, limites et autres simplifications | 6 |  |
| 3.3 | · Complexité informatique | 6 |  |
| 4 | Fonctionnalités de l'application: | | |
| 4.1 | · Fonctionnement général | 6 |  |
| 4.2 | · Entrées et sorties | 4 |  |
| 4.3 | · Maquettes | 15 |  |
| 4.4 | · Scénarios d'utilisation | 22 |  |
| 5 | Diagramme de classes UML préliminaire | 8 |  |
| 6 | Échéancier | 5 |  |
| 7 | Fonctionnalités optionnelles | 4 |  |
| 8 | Conclusion | 3 |  |
| **Sous total (avant déductions):** | |  |  |
| Déductions français: \_\_\_\_\_\_\_ fautes différentes | |  | ( - ) |
| Liste des sujets abordés mal remplie ou absente | |  | ( - ) |
| Format non conforme au gabarit, ou autres déductions: | |  | ( - ) |
| **Total:** | | **100** |  |
| **Appréciation des spécifications** | | | |
| ¨ Spécifications du projet suffisantes, l'équipe peut passer à l'étape du développement, en tenant compte des commentaires de correction.  ¨ Niveau de détails insuffisant pour établir la complexité du projet. Le développement doit être amorcé avec prudence. | | | |
| **Exigences uniformisées** | | | |
| ¨ L'application projetée tient compte de toutes les exigences uniformisées  ¨ L'application projetée a des lacunes concernant une ou plusieurs exigences uniformisées  ¨ Aspect scientifique en évidence et vérifiable  ¨ Triche/indices pour application à épreuve(s)  ¨ Capacité à vérifier les résultats  ¨ Capacité à réinitialiser  ¨ Présence du système d'aide uniformisé | | | |
| **Seuil de qualité / complément d’information à fournir** | | | |
| ¨ Soumission suffisamment claire et complète. Rien à remettre de nouveau.  ¨ Discussion requise pour clarifier certains éléments secondaires. Rien à remettre.  ¨ Discussion nécessaire pour clarifier certains éléments cruciaux. Il sera ensuite décidé si certaines sections devront être remises à nouveau.  ¨ Section(s) suivante(s) à remettre de nouveau : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | | |
| **Appréciation globale de la complexité** | | | |
| ¨ La complexité du projet soumis semble légèrement inférieure à la moyenne. L’utilisateur devra être emballé par le niveau d’interactivité et la présentation de l’application.  ¨ La complexité du projet soumis semble adéquate.  ¨ La complexité du projet soumis semble légèrement supérieure à la moyenne. Assurez-vous de rapidement vous attaquer aux éléments critiques et de revoir souvent vos priorités. | | | |

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |