

Université de la Manouba  
Ecole Nationale Des Sciences De L'informatique



## Rapport du Projet de Programmation

Sujet:  
Réalisation d'un jeu sérieux

Réalisé par:  
classe II1F

Les membres du groupe:  
Takali Mohamed Aziz  
Berrihi Elyes  
Ayachi Mayssa  
Khelif Mohamed Amine

Encadré par:  
Mr Boubakri Anis

**Année Universitaire 2021/2022**

---

---

## **Sommaire:**

<b>I. Introduction .....</b>	<b>3</b>
<b>II. .Étude préalable .....</b>	<b>4 - 6</b>
<b>I.1. Les Jeux sérieux.....</b>	<b>4</b>
<b>I.2. Jeu Choisi .....</b>	<b>4 - 6</b>
<b>III . Guide Du Joueur .....</b>	<b>;.... 7 - 9</b>
<b>IV. Guide du développeur .....</b>	<b>10 - 14</b>
<b>V. Etude préalable.....;</b>	<b>15 - 17</b>
<b>V.1. Langage et Technologies.....</b>	<b>15</b>
<b>V.2. Chronogramme et répartition des tâches.....</b>	<b>16 - 17</b>
<b>VI. Conclusion.....</b>	<b>18</b>

---

## I.Introduction:

Dans le cadre de notre première année à l'Ecole Nationale Des Sciences Informatiques (ENSI), nous avons eu comme tâche la réalisation d'un projet de programmation qui consiste à développer un jeu sérieux.

Nous avons constaté qu'une majorité des personnes n'est pas assez cultivée dans le domaine de l'astronomie.

Notre objectif consiste alors à introduire les curieux dans ce domaine et les faire vivre une expérience permettant de simuler les aventures d'un astronaute grâce à ce jeu 3D .

Dans ce rapport nous allons tout d'abord exposer l'analyse de ce jeu, puis un manuel d'utilisateur et enfin un manuel du programmeur.



---

## **II.Etude Préalable:**

### **II.1. Les jeux sérieux :**

Les jeux sérieux peuvent se définir comme des applications informatiques dont l'objet est de combiner à la fois des aspects sérieux

tels l'enseignement, l'apprentissage, la communication, avec des ressorts ludiques issus du jeu vidéo. Ainsi, ils favorisent l'immersion et peuvent faire appel à de nombreuses capacités telles que la résolution d'éénigmes, le développement de réflexes, le travail de mémorisation.

Il y a trois grandes catégories de jeu sérieux:

1- Les jeux éducatifs : servent à l'acquisition de connaissances ou l'augmentation des compétences.

L'apprenant se retrouve dans un scénario proche ou non de sa réalité.

2- Les jeux persuasifs : servent à passer des messages informatifs, persuasifs ou subjectifs, comme des publicités,

du marketing, des messages civiques.

3- Les jeux de simulation : qui reproduisent des activités réelles et servent à entraîner l'apprenant à l'acquisition de réflexes.

### **II.1. Jeu choisi :**

"Curious\_Astro" est un jeu sérieux. Il a un aspect pédagogique qui attise la curiosité du joueur. Il comprend notamment une partie de partage d'informations et également une partie d'évaluation. Le joueur enrichit ainsi ses connaissances dans le domaine de l'astronomie à partir d'une expérience interactive et compétitive.

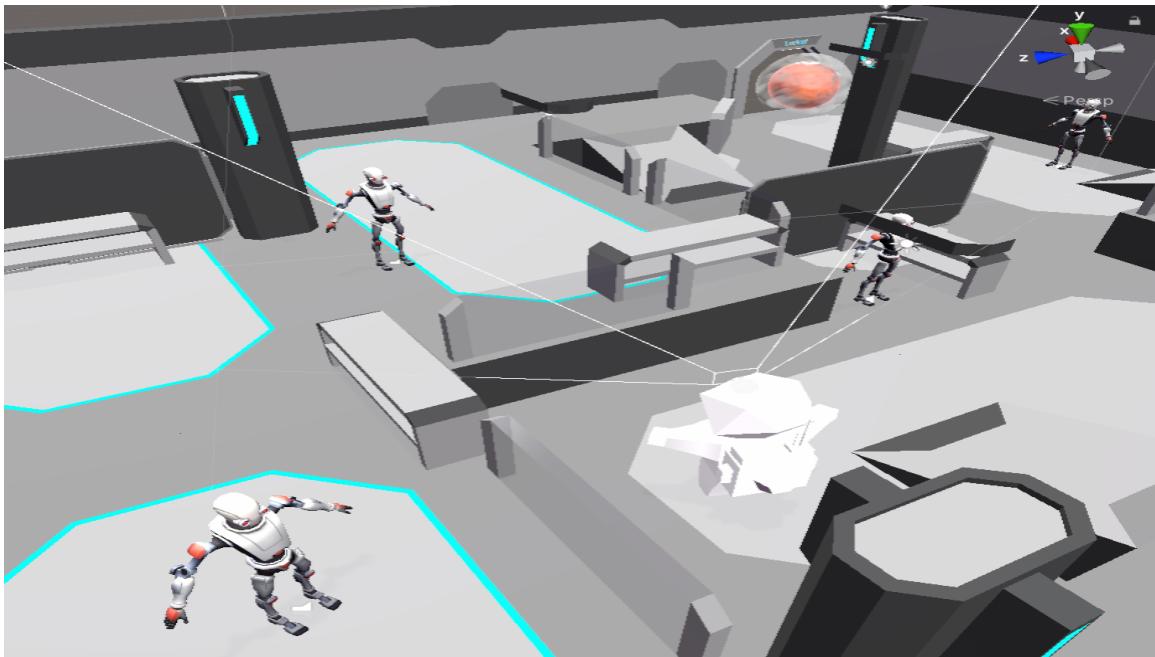
---

Le concept du jeu est le suivant :

Vous êtes un astronaute en mission de la NASA qui a perdu sa route en raison d'une panne technique de son vaisseau spatial. Vous êtes alors amené à gagner le jeu pour résoudre ces deux problèmes.

Vous avez 3 chapitres :

- **The Lab** : un chapitre introductif et facultatif où vous avez des informations générales sur l'espace, partagées par les robots de laboratoire de la NASA.



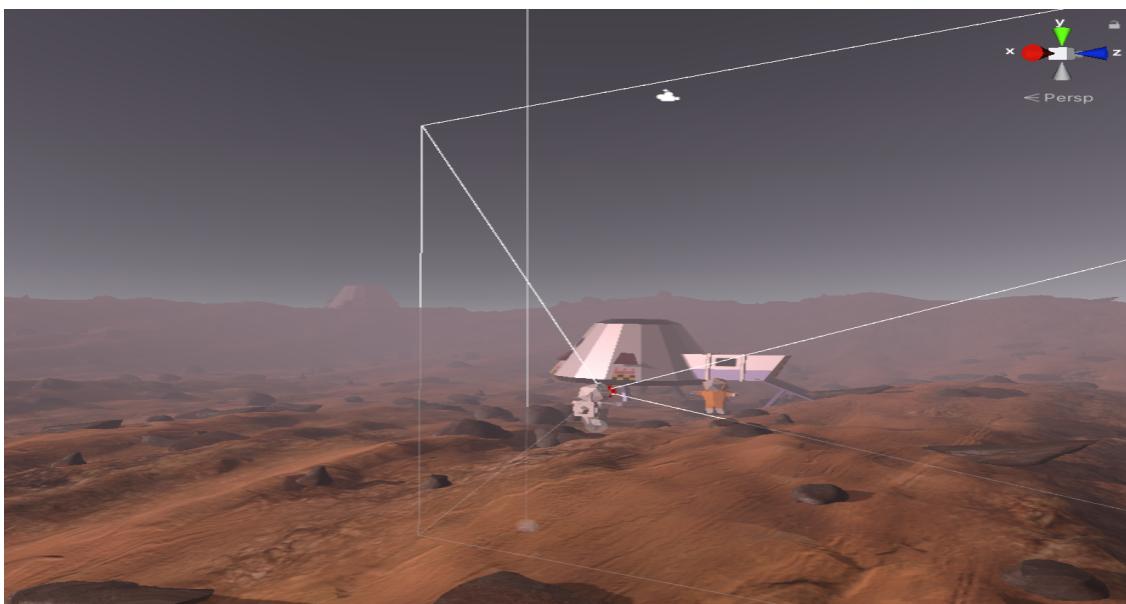
- **The Moon** : un chapitre obligatoire où l'on découvre de nouvelles informations sur ce satellite, partagées par d'autres astronautes envoyés en mission sur la Lune par la NASA.

=> Vous recevrez une carte de votre itinéraire lorsque vous gagnerez ce chapitre.



- **Mars** : un deuxième chapitre obligatoire où vous voyagerez dans le futur et recevrez des informations sur la planète rouge, partagées par les astronautes envoyés sur Mars pour la mission de l'année 2030 de la NASA.

=> Vous aurez un nouveau vaisseau spatial pour continuer votre mission.



---

### III. Guide du joueur :

#### Description des règles du jeu:

##### Le processus du jeu:

Dans un premier temps, notre curieux va se trouver dans une showroom qui présente une mise en situation dans le jeu.

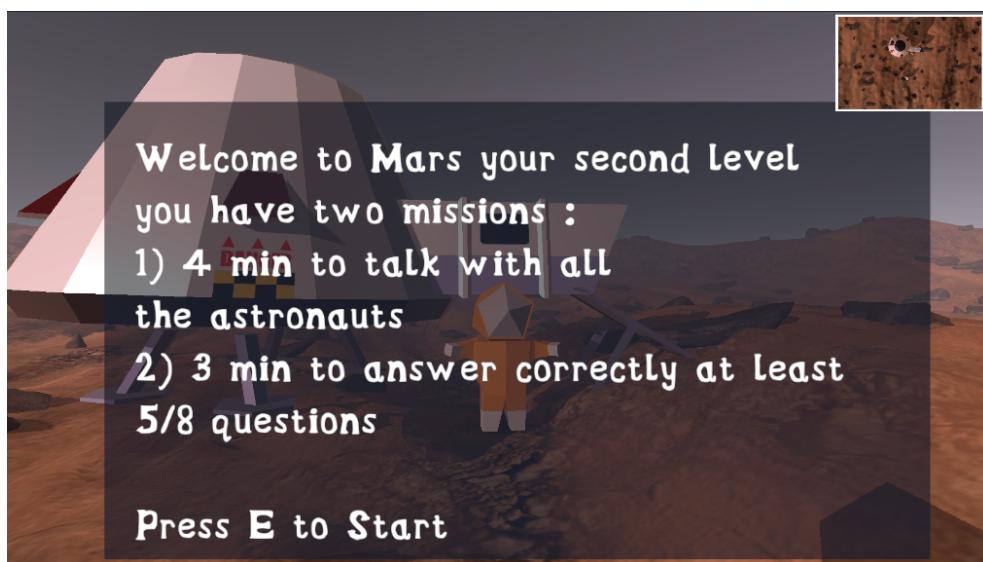
À partir de celle-ci, il aura uniquement l'accès à rejoindre le laboratoire où il discutera avec des robots afin d'acquérir des informations générales à propos de l'exploration spatiale.

De là, il aura accès pour aller découvrir la Lune, qui est l'unique chapitre accessible au début.

S'il réussit le stage de la lune, il rentrera au labo et le chapitre Mars devient accessible pour l'explorer.

=> S'il réussit ce dernier stage, le joueur gagne.

##### Que faire dans un chapitre ?



---

## **1ère mission:**

Dans chaque stage, il y a 5 astronautes que le joueur doit trouver et discuter avec eux dans une durée qui ne dépasse pas 4 minutes.

Le joueur doit retenir les informations fournies par ces astronautes



## 2ème mission:

Dès qu'il discute avec le dernier astronaute, 8 fusées apparaissent sur le terrain. Chacune représente une question avec un minuteur de 10 secondes.

Le joueur est considéré comme gagnant dans un stage s'il répond juste au moins sur 5 questions dans une durée qui ne dépasse pas 3 minutes.



## IV. Guide du développeur :

```
C# Dialogue.cs X
Assets > Scripts > C# Dialogue.cs > ...
● 1  using System.Collections;
  2  using System.Collections.Generic;
  3  using UnityEngine;
  4  using UnityEngine.UI;
  5
  6  public class Dialogue : MonoBehaviour {
  7
  8      public static int astroCounter;
  9      public string[] dialogue;
 10     public bool isTalking;
 11 }
```

=> [Dialogue.cs](#): Assurer le partage des informations entre les astronautes et le joueur dans les chapitres 'Mars' et 'The Moon'.

```
C# firstpersonlabo.cs X
Assets > Scripts > C# firstpersonlabo.cs > {} StarterAssets
 1  using UnityEngine;
 2  using UnityEngine.UI;
 3  using UnityEngine.SceneManagement;
 4  #if ENABLE_INPUT_SYSTEM && STARTER_ASSETS_PACKAGES_CHECKED
 5  using UnityEngine.InputSystem;
 6  #endif
 7
 8  namespace StarterAssets
 9  {
10      [RequireComponent(typeof(CharacterController))]
11      #if ENABLE_INPUT_SYSTEM && STARTER_ASSETS_PACKAGES_CHECKED
12      [RequireComponent(typeof(PlayerInput))]
13      #endif
14      0 references
15      public class firstpersonlabo : MonoBehaviour
16      {
17          [Header("Player")]
18          [Tooltip("Move speed of the character in m/s")]
19          1 reference
  }
```

=> [firstpersonlabo.cs](#): Configurer le déplacement du joueur dans le laboratoire et assurer le partage des informations entre les robots et le joueur.

```
C# GameConfig.cs X
Assets > Scripts > C# GameConfig.cs > ...
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.SceneManagement;
5  using UnityEngine.UI;
6  using UnityEngine.EventSystems;
7
8  namespace StarterAssets {
9      7 references
10     public class GameConfig : MonoBehaviour{
11         7 references
12         public static bool isMoon;
13         5 references
14         public static bool endChapterTime;
15         4 references
16         public static bool endFirstMission;
17
18 }
```

=> [GameConfig.cs](#): Manipuler les deux missions dans les deux chapitres ‘Mars’ et ‘The moon’

```
C# ItemCollector.cs X
Assets > Scripts > C# ItemCollector.cs > ...
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.UI;
5  #if ENABLE_INPUT_SYSTEM && STARTER_ASSETS_PACKAGES_CHECKED
6  using UnityEngine.InputSystem;
7  #endif
8  namespace StarterAssets {
9
10     4 references
11     public class ItemCollector : MonoBehaviour{
12         5 references
13         public static int visitedRockets;
14         9 references
15         public static int score;
16
17 }
```

=> [ItemCollector.cs](#): Gérer les quiz et leurs réglages + calculer le score .

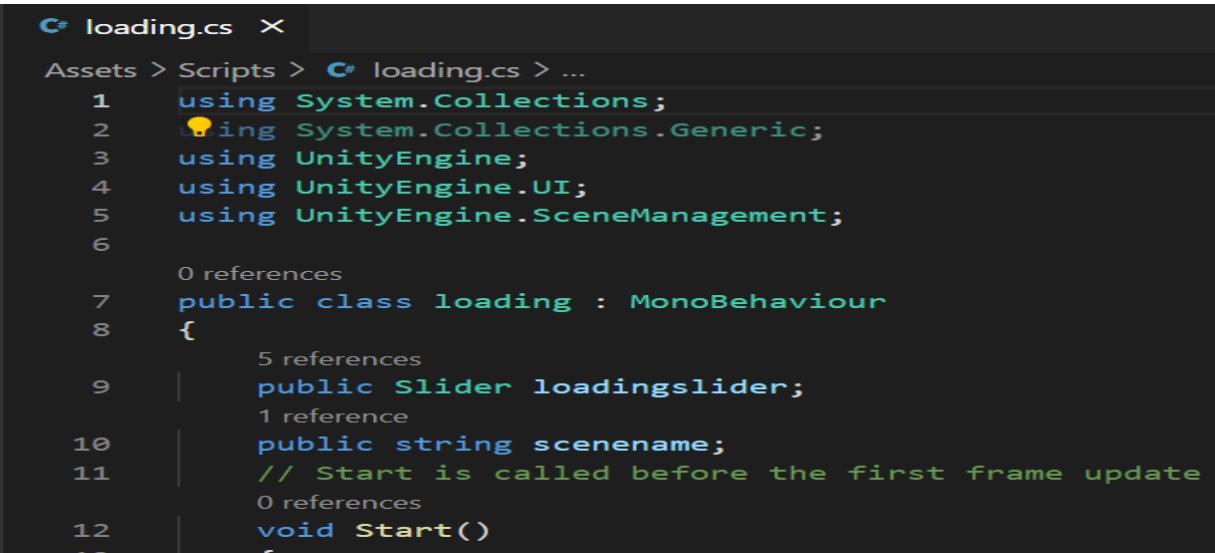
```
C# FirstPersonController.cs X
Assets > Scripts > C# FirstPersonController.cs > ...
1  using UnityEngine;
2  using UnityEngine.UI;
3  #if ENABLE_INPUT_SYSTEM && STARTER_ASSETS_PACKAGES_CHECKED
4  using UnityEngine.InputSystem;
5  #endif
6
7  namespace StarterAssets
8  {
9      [RequireComponent(typeof(CharacterController))]
10     #if ENABLE_INPUT_SYSTEM && STARTER_ASSETS_PACKAGES_CHECKED
11     [RequireComponent(typeof(PlayerInput))]
12     #endif
13     10 references
14     public class FirstPersonController : MonoBehaviour
15     {
16         [Header("Player")]
17         [Tooltip("Move speed of the character in m/s")]
18         1 reference
19         public float MoveSpeed = 4.0f;
```

=> [FirstPersonController.cs](#): Configurer le déplacement du joueur dans les deux chapitres ‘Mars’ et ‘The moon’.

```
C# sceneswitcher.cs X
Assets > Scripts > C# sceneswitcher.cs > ...
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.SceneManagement;
5  using UnityEngine.UI;
6
7  namespace StarterAssets{
8      0 references
9      public class sceneswitcher : MonoBehaviour
10     {
11         0 references
12         public void playGame()
```

=> [scenceswitcher.cs](#): Configurer le passage entre les scènes 2D et 3D.

---

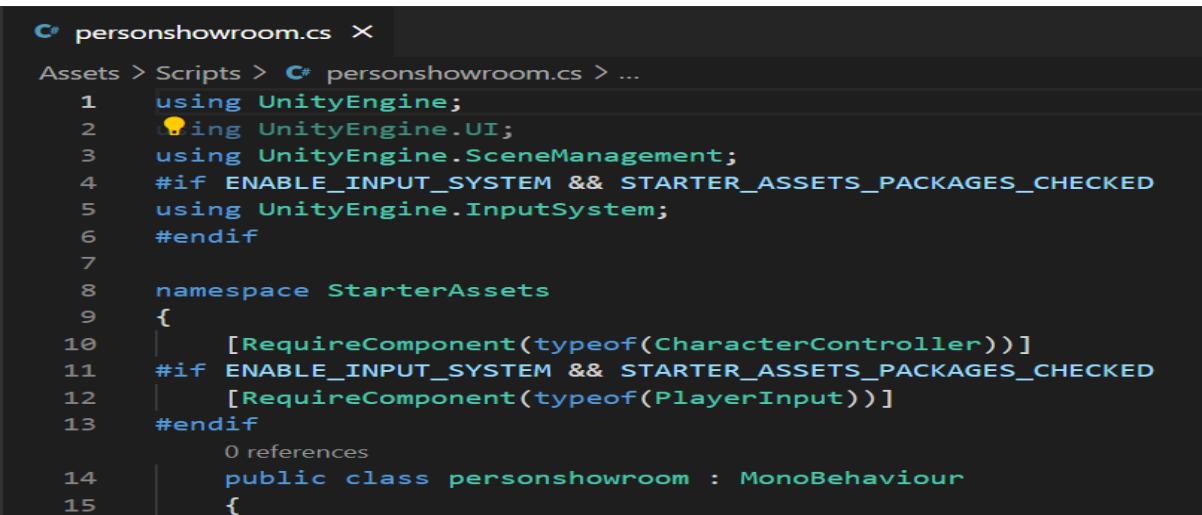


```
C# loading.cs ×

Assets > Scripts > C# loading.cs > ...

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.UI;
5  using UnityEngine.SceneManagement;
6
7  0 references
8  public class loading : MonoBehaviour
9  {
10     5 references
11     public Slider loadingslider;
12     1 reference
13     public string scenename;
14     // Start is called before the first frame update
15     0 references
16     void Start()
17     {
18         loadingslider.value = 0;
19     }
20 }
```

=> [loading.cs](#): Le script de la scène ‘Loading’



```
C# personshowroom.cs ×

Assets > Scripts > C# personshowroom.cs > ...

1  using UnityEngine;
2  using UnityEngine.UI;
3  using UnityEngine.SceneManagement;
4  #if ENABLE_INPUT_SYSTEM && STARTER_ASSETS_PACKAGES_CHECKED
5  using UnityEngine.InputSystem;
6  #endif
7
8  namespace StarterAssets
9  {
10     [RequireComponent(typeof(CharacterController))]
11     #if ENABLE_INPUT_SYSTEM && STARTER_ASSETS_PACKAGES_CHECKED
12     [RequireComponent(typeof(PlayerInput))]
13     #endif
14     0 references
15     public class personshowroom : MonoBehaviour
16     {
```

=> [personshowroom.cs](#): Configurer le déplacement du joueur dans la showroom.

---

```
C# scenemanager.cs X
Assets > Scripts > C# scenemanager.cs > ...
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.SceneManagement;
5
6  public class scenemanager : MonoBehaviour
7  {
8      public void LoadScene(string scenename)
9      {
10          SceneManager.LoadScene(scenename);
11      }
12  }
13
```

=>[scenemanager.cs](#): Configurer le passage entre les scènes.

---

## V.Etude Préalable :

### V.1. Langages et technologies:

#### Pourquoi Unity ?

Il est l'un des plus répandus dans l'industrie du jeu vidéo. Unity facilite le travail et le rend plus rapide grâce à ses différentes fonctionnalités scripts prédéfinies .

Concernant la partie conception, les assets gratuits offerts par Unity qui nous ont facilités différentes tâches sans oublier la mise en place des scènes souhaitées .

En plus , il est plus pratique de l'utiliser suite à l'existence des forums et des vidéos en ligne vu la popularité de cette interface qui nous a permis d'acquérir les informations nécessaires et pouvoir résoudre les problèmes vécus .

#### Pourquoi C# ?

Vu notre Parcours académique suivi , et pour faire adapter la partie du code avec l'interface du jeu, on a choisi C#. Étant le langage le plus générique et performant en termes de rapidité de traitement pour le développement des jeux, il nous a donné un plus grand contrôle au niveau de la conception.

#### Pourquoi Github ?

GitHub offre un service de cloud qui aide les développeurs à stocker et à gérer leur code, ainsi qu'à suivre et contrôler les modifications qui lui sont apportées. Donc grâce à cette plateforme on a pu collecté en total et géré le reste en équipe plus aisément .



---

## **V.2. Chronogramme : Répartition des tâches :**

Nom du tâche	Responsable	WEEK 1 27/02	WEEK 2 06/03	WEEK 3 13/03	WEEK 4 20/03	WEEK 5 27/03	WEEK 6 03/04	WEEK 7 10/04	WEEK 8 17/04	WEEK 9 24/04	WEEK 10 01/05
<b>MENU ET LES SCENES 2D INTERMEDIAIRES</b>											
Design	AZIZ										
Liaison avec les autres scenes	MAYSSA										
<b>SHOWROOM</b>											
Design	MAYSSA										
Mise en place d'un vocal indicatif	AMINE										
Liaison avec la scène du laboratoire	AZIZ										
Développement du script personshowroom	ELYES										
<b>LABORATOIRE</b>											
Design	MAYSSA										
Mise en place des vox indicatifs											
inindicatif	ELYES										
Liaison avec la scène de la lune											
	MAYSSA										
Liaison avec la scène de la planète Mars	AZIZ										
Développement du script firstpersonlabo	ELYES										
<b>MARS</b>											
Design	AMINE										
Développement du script FirstPersonController	ELYES										
Début du développement du script GameConfig	AMINE										
Mise en place d'un minimap	AMINE										

Mission1:collecter les données à partir des astronautes:											
Mise en place des astronautes sur scène	AZIZ										
Développement du script Dialogue	MAYSSA										
Choix des informations mises dans les dialogues	MAYSSA										
Mise en place d'un minuteur	ELYES										
<b>Mission2:répondre aux quiz</b>											
Mise en place des roquettes sur scène	AZIZ										
Développement du script ItemCollector	ELYES										
Mise en place des questions	AMINE										
Mise en place des minuteurs											
<i>Minuteur1:pour toute la mission</i>	AZIZ										
<i>Minuteur2:pour chaque question</i>	MAYSSA										
Liaison avec les autres scènes	AZIZ										
<b>LUNE</b>											
Design	ELYES										
Mise en place d'un minimap	AMINE										
<b>Mission1:collecter les données à partir des astronautes:</b>											
Mise en place des astronautes sur scène	AZIZ										
Actualisation du script Dialogue	AMINE										
Choix des informations mises dans les dialogues	AZIZ										
Mise en place d'un minuteur	AMINE										
<b>Mission2:répondre aux quizz</b>											
Mise en place des roquettes sur scène	MAYSSA										
Actualisation du script ItemCollector	ELYES										
Mise en place des questions	ELYES										
Mise en place des minuteurs											
<i>Minuteur1:pour toute la mission</i>	ELYES										
<i>Minuteur2:pour chaque question</i>	MAYSSA										
Liaison avec les autres scènes	AMINE										
<b>FINITION</b>	TOUS										

---

## **VI. Conclusion :**

### **Les difficultés rencontrées :**

- Au début et en tant que débutants, nous avons rencontré des difficultés concernant la bonne utilisation du moteur Unity et du langage C#. Cela nous a pris un peu de temps, notamment dans la partie implémentation des classes.
- Les actifs gratuits sur le site Unity ne sont pas très variés. Nous avons donc dû construire nous-mêmes la scène du chapitre "La Lune".

### **Améliorations futures possibles :**

- Nous pouvons ajouter d'autres chapitres dans d'autres planètes. (Ce n'était pas possible pour le moment en raison du manque d'actifs libres)
- Vous pouvez ajouter des sous-titres pour les dialogues des chapitres 'Showroom' et 'The Lab'.