



Oh-oh-Ho-ho... in the shadows!

Projet Unity

Ly [ly@staff.42.fr](mailto:ly@staff.42.fr)  
42 Staff [pedago@staff.42.fr](mailto:pedago@staff.42.fr)

*Résumé: Ce projet a pour but de vous faire coder un jeu complet en utilisant Unity.*

# Table des matières

<b>I</b>	<b>Préambule</b>	<b>2</b>
<b>II</b>	<b>Introduction</b>	<b>4</b>
<b>III</b>	<b>Objectifs</b>	<b>5</b>
<b>IV</b>	<b>Consignes générales</b>	<b>6</b>
<b>V</b>	<b>Partie obligatoire</b>	<b>7</b>
V.1	Le jeu . . . . .	7
V.1.1	Menu . . . . .	7
V.1.2	Gameplay . . . . .	8
V.1.3	Différents niveaux de difficulté . . . . .	9
<b>VI</b>	<b>Partie bonus</b>	<b>10</b>
<b>VII</b>	<b>Rendu et peer-évaluation</b>	<b>11</b>

# Chapitre I

## Préambule

Le préambule a un vague rapport avec ce que vous allez devoir faire :

No sleep  
No sleep, until I'm done with finding the answer  
Won't stop  
Won't stop before I find the cure for this cancer

Sometimes  
I feel like going down, I'm so disconnected  
Somehow  
I know that I am haunted to be wanted

I've been watching, I've been waiting  
In the shadows for my time  
I've been searching, I've been living  
For tomorrows all my life

Oh oh oh oh  
In the shadows  
Oh oh oh oh  
In the shadows

They say  
That I must learn to kill before I can feel safe  
But I,  
I'd rather kill myself than turn into their slave

Sometimes  
I feel that I should go and play with the thunder  
Somehow  
I just don't wanna stay and wait for a wonder

I've been watching, I've been waiting  
In the shadows for my time  
I've been searching, I've been living

For tomorrow's all my life

Lately, I've been walking, walking in circles  
Watching, waiting for something  
Feel me, touch me, heal me, come take me higher

I've been watching, I've been waiting  
In the shadows for my time  
I've been searching, I've been living  
For tomorrow's all my life

I've been watching  
I've been waiting  
I've been searching  
I've been living  
For tomorrow's

Oh oh oh oh  
In the shadows  
Oh oh oh oh  
In the shadows  
I've been waiting

In the shadows - The Rasmus

# Chapitre II

## Introduction

Dans ce projet, nous allons développer un jeu ressemblant à Shadowmatic. Le but du jeu est le suivant : on doit reproduire une forme reconnaissable en ombre sur un mur, à l'aide d'objets situés au premier plan.

Rendez-vous sur la page du jeu pour mieux comprendre :

<http://www.shadowmatic.com/>

Et si vous avez un Iphone, le mieux, c'est que vous la téléchargez pour tester !

# Chapitre III

## Objectifs

L'objectif de ce projet est de vous faire implémenter un jeu simple de A à Z en utilisant Unity.  
Le game design n'est pas très complexe, mais nécessite un peu de créativité.

# Chapitre IV

## Consignes générales

- Ce projet est un projet Unity, il est donc à faire entièrement et obligatoirement en C# uniquement. Pas de Javascript/Unityscript, de Boo ou autres horreurs.
- Au cas où vous vous poseriez la question, il n'y a pas de norme imposée à 42 pour le C#.  
Vous pouvez utiliser le style qui vous plaît sans restriction. Mais rappelez-vous qu'un code que votre peer-évaluateur ne peut pas lire est un code qu'elle ou il ne peut noter.
- Vous devez trier les assets de votre projet par dossier. Chaque dossier correspond à un et un seul type d'asset. Par exemple : "Scripts/", "Scenes/", "Sprites/", "Prefabs/", "Sounds/", "Models/", ...
- L'utilisation de l'Asset Store d'Unity est interdite. Vous êtes encouragés à chercher des assets sur le net, sauf bien entendu pour les scripts car vous devez avoir écrit tout ce que vous rendez. L'utilisation des Standard Assets de Unity est autorisée.
- Les corrections se feront en peer-correcting.
- Pour la correction, il vous sera demandé de build le jeu pour le tester. C'est le correcteur qui doit build le jeu vous devez donc évidemment push tout ce qui est nécessaire à ce build. De ce fait votre projet doit être correctement configuré pour le build. Aucun réglage de dernière minute ne doit être toléré.

# Chapitre V

## Partie obligatoire

### V.1 Le jeu

Dans le jeu Shadowmatic initial, il y a un ou plusieurs objets situés au premier plan. On peut faire bouger ces objets par rotation. Le but est de reproduire avec l'ombre des objets une forme reconnaissable sur le mur situé en arrière plan.

#### V.1.1 Menu

Votre jeu doit afficher au démarrage un menu permettant de lancer le jeu dans deux modes différents :

- Le mode normal : redirigera le joueur vers le chemin contenant les différents puzzles. On doit voir distinctement la différence entre un puzzle résolu, un puzzle débloqué mais non résolu et un puzzle non débloqué.
- Le mode testeur : redirigera le joueur vers le chemin contenant les différents puzzles qui seront tous débloqués afin de pouvoir tester aisément en soutenance.

Cela signifie aussi que l'avancement du joueur doit être sauvegardé. Pour la partie obligatoire, une sauvegarde par "appareil" est suffisante.

Vos puzzles doivent avoir un nom qui donne un indice sur l'ombre à réaliser. Comme dans le jeu original, vous ne devez pas montrer au joueur ce qu'il doit faire.



Un puzzle qui s'appelle "Meow" donnera raisonnablement une ombre ressemblant à un chat.



## V.1.2 Gameplay

Le jeu doit se jouer entièrement à la souris. On autorisera l'association du clic de la souris avec une touche du clavier (par exemple, CTRL) pour différencier les mouvements que l'on applique aux objets.

Votre jeu doit être jouable. C'est-à-dire que la gestion de la validation de l'ombre doit n'être ni trop exigeante, ni trop laxiste. Si la validation se fait au pixel près, ça ne va pas, s'il y a 50 pixels d'écart, ça ne va pas non plus.

Quand vous considérez l'ombre sur le mur comme étant valable, un message apparaît pour auréoler le joueur de gloire et un menu lui permettra soit de quitter, soit de revenir au menu dont il vient, pour contempler la magnifique animation que vous ferez lorsqu'un puzzle se valide et qu'un autre se débloque.

Le déplacement des objets doit être fluide.

### V.1.3 Différents niveaux de difficulté

Votre jeu devra comporter 3 niveaux de difficulté, appliqué sur différents puzzles :

1. Un seul objet auquel on peut seulement appliquer des rotations horizontales.
2. Un seul objet auquel on peut appliquer des rotations horizontales et verticales.
3. Plusieurs objets auxquels on peut appliquer des rotations horizontales et verticales et qu'on peut déplacer dans l'espace.

C'est pour différencier la rotation du déplacement que vous pouvez combiner le clic souris avec une touche du clavier.



Clic & drag : je fais tourner l'objet horizontalement. (Clic + CTRL)  
& drag : je fais tourner l'objet verticalement. (Clic + Maj) & drag :  
je déplace mes objets.

Vous devez réaliser au moins un puzzle de chaque niveau.

Des assets vous sont fournis pour réaliser les puzzles obligatoires, mais libre à vous d'en utiliser d'autres.

# Chapitre VI

## Partie bonus

Nous considérerons 4 bonus différents, et ce uniquement si vous avez **ENTIÈREMENT** complété la partie obligatoire du projet :

- La beauté : Vous avez implémenté de jolies textures, votre menu claque, le chemin de puzzle est bien fait... Bref, votre jeu est trop bô.
- La créativité : Vous avez implémenté plus d'ombres que demandées dans la partie obligatoire.
- L'ambiance : Vous avez ajouté une super bande son à votre jeu, et avez permis au joueur d'activer ou non cette bande son via un menu d'options, qui peut aussi contenir d'autres options (jeu au clavier, taille de fenêtre...)

# Chapitre VII

## Rendu et peer-évaluation

Rendez-votre travail sur votre dépôt GiT comme d'habitude. Seul le travail présent sur votre dépôt sera évalué en soutenance.