



TECH

CAHIER DES CHARGES

Les bases du C#



INTRODUCTION

Nom du projet : Les bases du C#

Projet N° :

Date :

Classe : G.Tech 2

MODALITÉS

Travail et projet par groupe de 3 étudiants ; évaluation en groupe sur le projet.

Pour la communication écrite et les coachings en présentiel , utilisation de Discord.

CONTEXTE ET DESCRIPTION DU PROJET

Le but de ce projet est de découvrir et mettre en place les différents principes de la programmation orienté objet (POO) en produisant un jeu en console avec C#.

Ce jeu devra comporter des combats en tour par tour à la manière des jeux Pokémon ou les premiers Final Fantasy (attaque, utilisation d'objet, magie)
Pas de graphisme à intégrer, le jeu se jouera avec le terminal et comportera donc une vue TopDown avec une carte en ASCII.

Ce jeu devra comporter :

Une carte où le joueur pourra se déplacer librement à l'aide des touches fléchées
Des combats aléatoires sur certaines cases spécifiques (équivalent à l'herbe haute de Pokémon)

Un menu où l'on peut voir notre équipe, consulter leur statistiques et attaques possibles.

Un menu où l'on peut consulter notre inventaire (potion, clés, objet de boost de statistiques)

Un menu permettant la sauvegarde et le chargement de notre partie (sérialisation).

Les combats doivent :

proposer un affrontement en 1v1 au tour par tour

L'ennemi sera piloté par une IA.

Si notre personnage tombe au combat on peut le remplacer par un autre membre de notre équipe, si tous nos personnages sont KO : GameOver

À chaque tour nous devons choisir notre mouvement (attaque, magie, objet), chaque menu présente un ensemble de mouvement possible et le joueur doit choisir le mouvement précis à utiliser. La magie consomme une barre de mana qui se recharge progressivement.

Chaque personnage comporte un type, des statistiques (attaque, défense, vitesse, PV, PM, précision etc) et un set d'attaque et de magie qui lui appartient. Les personnages évoluent grâce à de l'expérience acquise en combat et peuvent monter de niveau (meilleures statistiques et potentiellement nouvelle attaque).

Chaque mouvement comporte un type, des statistiques d'attaque, de précisions, taux coup critique etc. Les types sont efficaces face à d'autres types ou au contraire faible face à d'autres (voir le tableau des types de Pokémon).

En cas de victoire nos personnages gagnent de l'expérience.

Le jeu évolue en 2 phases :

La carte du jeu, où notre personnage se déplace sur une carte à l'aide des touches fléchées. Sur cette carte on peut parler à des personnages, ouvrir des coffres, enclencher des combats avec des ennemis.

Les combats, phase stratégique du jeu exploitant l'ensemble des mécaniques d'affrontement comme décrit plus haut.

Il sera important de réfléchir à l'avance de la structure du jeu et de ses différents composants. Pour cela on produira un ensemble de diagramme de classes afin d'avoir une meilleure vision du projet. On pourra exploiter l'héritage et tout ce qui vient avec au travers des différents personnages, les attaques, les objets etc.

Les combats étant majoritairement une application mathématique on mettra en place un ensemble de tests unitaires pour valider la cohérence et la solidité du système.

Ceci est une première approche de ce projet, des altérations sont possibles si c'est demandé et validé par l'intervenant au cours du projet.

SPECS

Vous devez implémenter dans votre application :

- Système de carte TopDown
- Système de combat
- Sauvegarde & chargement

Pour les plus avancés uniquement (ou pour aller plus loin)

- Implémentation de son
- Mécaniques d'IA, d'animation, quêtes, dialogues etc.

Informations complémentaires :



OBJECTIFS PÉDAGOGIQUES ET PROFESSIONNELS DU PROJET

A l'issue de ce projet, les étudiants seront capables de

Savoirs

- Chercher de la documentation
- Prioriser les tâches
- Découper l'organisation pour travailler plus efficacement à plusieurs
- Identifier les systèmes nécessitant des tests unitaires
- Définir une architecture modulaire et extensible

Savoir-être

- Travailler effectivement en groupe
- Planifier le bon déroulement du projet
- Respecter les délais
- Évaluer sa propre contribution
- Reconnaître la contribution des autres

Savoir-faire / Compétences

- Programmer en C# avec une approche POO
- Créer une application console de façon fluide
- Appliquer les règles de bonnes pratiques générales à la programmation.

RESSOURCES

1. **A savoir/à apprendre** (Ce travail pourra être évalué / noté en début de projet) :

- Règles de la programmation
- Principes de bases de la POO
- Explorer ce qui est disponible / possible en console

2. A lire/à consulter :

- Documentation MSDN Console :

<https://learn.microsoft.com/fr-fr/dotnet/api/system.console?view=net-7.0>

POO

<https://learn.microsoft.com/fr-fr/dotnet/csharp/fundamentals/tutorials/oop>

Tests Unitaires :

<https://learn.microsoft.com/fr-fr/visualstudio/test/unit-test-basics?view=vs-2022>

3. A installer/pré-requis techniques :

- Télécharger Visual Studio avec le module Développement .Net Desktop le composant individuel “Concepteur de classes” et préparer le repo Git.

TRAVAIL PRÉPARATOIRE

CONSTITUER DES GROUPES DE 3 ÉTUDIANTS.

RÉFLÉCHIR AU DESIGN GÉNÉRAL DU JEU

SE DOCUMENTER SUR LA PROGRAMMATION ORIENTÉ OBJET, L'HÉRITAGE, VIRTUAL/OVERRIDE ET L'API
CONSOLE DU C# .NET



INTERVENANT



Nom : Kévin Roussel

Titre : Développeur spécialisé Unity, gérant de Red Studio

Email : lebruitdered@gmail.com

Site web :

LinkedIn : <https://www.linkedin.com/in/kevinroussel/>

BIOGRAPHIE

Gérant de la société RedStudio, studio de production de jeux vidéo depuis 2019. La société a pour but de produire et de vulgariser la création de jeux notamment au travers d'une chaîne Youtube et Twitch. Les jeux produits sont la base des épisodes de vulgarisation. J'interviens également en école spécialisé en JV dans toute la France, principalement sur le développement avec Unity mais également en abordant des thématiques plus transversales comme la modélisation avec Blender.



ROADMAP

DESCRIPTION

Veillez trouver ci-dessous une description des livrables attendus ainsi que les dates d'échéance associées. Il est essentiel de respecter les échéances suivantes pour assurer une progression harmonieuse et structurée du projet. Chaque livrable représente une étape importante dans le processus de réalisation et permet d'évaluer l'avancement du travail. Les dates limites fixées doivent être rigoureusement respectées afin de garantir une évaluation équitable et de permettre un feedback constructif en temps opportun.

Jalon	Livrables attendus	Date limite	Moyens / formats
1	Travail préparatoire	29/09/2025	GDoc de présentation du projet et de ses mécaniques / scope
2	Projet final	03/10/2025	Soumission à un formulaire fournis avec : Lien Github, vidéo de gameplay, build et toute documentation nécessaire
3	Bilan du projet	03/10/2025	Soutenance de projet

Merci de remplir le questionnaire feedback en fin de projet.



EVALUATION

L'évaluation est conçue pour être holistique, prenant en compte non seulement le produit final, mais aussi le processus, les compétences acquises et les attitudes démontrées tout au long du projet. Les compétences sont classées en trois catégories principales : “savoirs” (connaissances théoriques), “savoir-faire” (compétences pratiques), et “savoir-être” (compétences interpersonnelles et attitudes).

SYSTÈME DE NOTATION

Savoirs (Connaissances)

- **Compréhension théorique** : Évaluée soit en amont du projet pendant la semaine théorie soit lors de la soutenance et restitution du projet. Cela permet de mesurer la compréhension des concepts fondamentaux et des connaissances liées au projet des étudiants.

Savoir-faire (Compétences)

- **Compétences techniques et application** : Évaluées à travers la soumission finale du projet. Cela inclut la qualité, la fonctionnalité, et la précision technique du travail produit.
- **Gestion de projet** : Évaluée en fonction de l'organisation, du respect des échéances, et de l'utilisation efficace des ressources. Cela peut être évalué à travers la documentation du projet et les journaux de processus.

Savoir-être (Attitudes/Compétences interpersonnelles)

- **Travail d'équipe et collaboration** : Évalués à travers des évaluations par les pairs et les membres du groupe. Les critères incluent la communication, la coopération, et la contribution aux tâches du groupe.
- **Autonomie et initiative** : Évaluées en fonction des contributions individuelles, de la capacité à travailler de manière autonome, et de la résolution proactive des problèmes.



GRILLE D'ÉVALUATION

La grille d'évaluation du projet devra impérativement reprendre les savoirs/savoir-être/savoir-faire qui ont été cités à la partie Objectifs pédagogiques. Vous êtes libre de déterminer combien de points à accorder à chaque connaissance/compétence, mais la note finale doit être sur 20.

Connaissances / Compétences / Fonctionnalités	Barème
Qualité du code / bugs	/ 4
Tests unitaires	/ 2
Diagramme de classe	/ 2
Ecran de carte du jeu	/ 3
Combats	/ 3
Sauvegarde	/ 2
Fun factor (intérêt et qualité globale du jeu)	/ 2
Fonctionnalité(s) supplémentaire(s)	/ 2
TOTAL	/ 20

EXAMEN INDIVIDUEL DE FIN DE PÉRIODE

Notez ici les éléments relatifs au partiel : acquis à valider / modalités envisagées

Développement général en C# pour console et library class. Tests Unitaires.

.....
.....
.....
.....