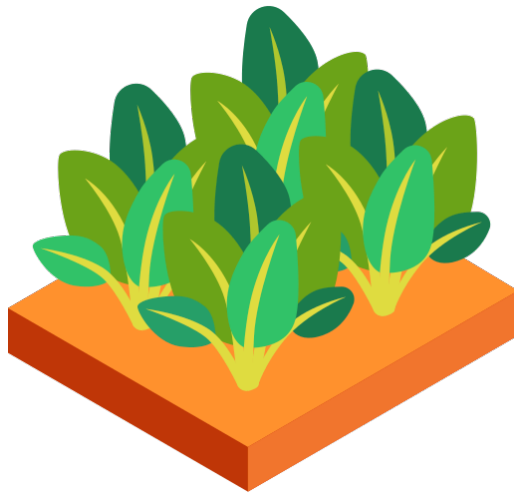


Projet ENSemenC - Rapport technique



PROGRAMMATION AVANCÉE

AMMAR-BOUDJELAL LINA

GROUPE 4

PROMOTION 2027

Sommaire

Introduction	4
1. L'univers du jeu et ses spécificités	5
1.1. Présentation générale de l'univers	5
1.2. Les mondes disponibles	5
1.3. Typologie des plantes	7
1.4. Facteurs environnementaux	9
1.5. Double temporalité du jeu	9
2. Déroulement d'une partie	11
2.1. Démarrage	11
2.2. Menu d'actions et interface du jeu	12
2.3. Tour de jeu hebdomadaire (mode classique)	14
2.4. Mode urgence	15
2.5. Fin de partie	16
3. Conception orientée objet	16
3.1. Structure générale du projet	16
3.2. Hiérarchie des classes	17
3.3. Diagramme UML	21
4. Tests et vérifications	21
4.1. Stratégie de test adoptée	21
4.2. Vérification des fonctionnalités principales	22
4.3. Cas particuliers testés	22
5. Gestion de projet	23

5.1. Contexte du projet	23
5.2. Inspirations créatives	23
5.3. Organisation	24
6. Bilan critique	25
Annexes	27
Annexe 1 : Diagramme UML général des classes du projet	27
Annexe 2 : Description initiale des 5 mondes du jeu	28

Introduction

Le projet ENSemenC est un jeu de simulation de potager développé en C# dans le cadre du cours de Programmation Avancée. Il propose au joueur de cultiver différentes plantes sur deux terrains de culture, en prenant en compte des facteurs environnementaux tels que la météo, la nature du sol, la température ou encore l'ensoleillement. Le jeu se déroule dans un univers fictif composé de deux mondes (Plaine Champignon et Désert Chomp), chacun ayant ses propres types de plantes, conditions de culture et nuisibles spécifiques. Le jeu inclut une double temporalité : un mode classique, où chaque tour représente une semaine de culture, et un mode urgence, où le joueur doit réagir rapidement à l'arrivée d'un nuisible menaçant ses plantes.

Le projet a été développé progressivement en commençant par une version minimale fonctionnelle, incluant les mécanismes de base comme le semis, l'arrosage, la récolte et la météo. Des fonctionnalités supplémentaires ont ensuite été ajoutées, telles que les maladies, les urgences, ou encore l'affichage stylisé en console.

Afin de garantir une lisibilité optimale et une facilité de maintenance, le code a été structuré en respectant le principe de séparation des responsabilités. Les différentes fonctionnalités du jeu sont réparties dans des fonctions dédiées, ce qui facilite leur compréhension et leur éventuelle modification. Enfin, le projet a été conçu pour rester évolutif, permettant l'ajout futur de nouvelles règles, fonctionnalités, ou même de mondes.

Ce rapport a pour objectif de détailler les choix techniques effectués, les méthodes employés pour réaliser le jeu et garantir son fonctionnement, ainsi que l'organisation adoptée au cours du développement.

1. L'univers du jeu et ses spécificités

1.1. Présentation générale de l'univers

ENSemenC est un jeu de simulation dans un univers fictif où le joueur incarne un jardinier en charge de deux terrains de culture. L'objectif est de faire prospérer un potager en semant, entretenant et récoltant diverses plantes, tout en faisant face à des contraintes environnementales et à des événements imprévus.

L'univers du jeu est inspiré du jeu vidéo Mario et organisé en mondes thématiques, chacun ayant ses propriétés environnementales, ses types de plantes spécifiques et ses nuisibles. Le joueur peut choisir dans quel monde il souhaite jouer en début de partie, ce qui influencera fortement ses stratégies de culture.

La progression du jeu repose sur une double temporalité :

- Un mode classique, où le temps passe semaine par semaine, avec des actions de gestion régulière (semis, arrosage, désherbage, etc.);
- Un mode urgence, qui se déclenche aléatoirement et dans lequel le joueur doit réagir rapidement face à une menace spécifique (nuisible, intempérie, etc.).

1.2. Les mondes disponibles

Le jeu propose actuellement deux mondes jouables, chacun avec son ambiance, ses règles environnementales et ses défis. Les mondes sont inspirés du jeu vidéo New Super Mario Bros.

1.2.1. Monde 1 - Plaine Champignon

La Plaine Champignon est inspiré du monde 1 de New Super Marios Bros. Il s'agit d'un monde tempéré, inspiré des décors naturels et paisibles. C'est un environnement favorable pour débiter, avec des conditions climatiques modérées et une météo globalement équilibrée.

- **Température moyenne** : entre 10°C et 25°C.
- **Sols disponibles** : Humifère.
- **Plantes cultivables** : Tomates, fraises, marguerites et champignons.
- **Nuisibles spécifiques** :
 - Goombaver : attaque et dévore les plantes.
 - PuceronKoopas : volent les productions des plantes adultes.
 - Koopascargot : ralentit la croissance des jeunes plantes.

Ce monde représente un environnement équilibré, où la bonne gestion de l'eau suffit généralement à garantir une bonne récolte, à condition de surveiller l'apparition des nuisibles.

1.2.2. Monde 2 - Désert Chomp

Le Désert Chomp est un environnement beaucoup plus extrême, inspiré des régions désertiques et du monde 2 de New Super Mario Bros. Il impose des conditions arides et un ensoleillement très fort, mettant à l'épreuve la résistance des plantes.

- **Température moyenne** : entre 30°C et 45°C.
- **Sols disponibles** : Sableux.
- **Plantes cultivables** : Cactus, aloé vera, plantes feu et dattiers.
- **Nuisibles spécifiques** :
 - Chomp des sables : attaque les racines et déterre les plantes.
 - Scaraboss : ralentit la croissance de manière durable.

- **Tempête de sable** : événement météorologique extrême pouvant détruire les plantes les plus fragiles.

1.3. Typologie des plantes

Les plantes sont au coeur du jeu ENSemenC. Chaque type de plante possède des caractéristiques propres influençant sa croissance, sa résistance aux conditions climatiques, son espérance de vie, sa productivité, ainsi que sa sensibilité aux maladies.

1.3.1. Caractéristiques générale

Chaque plante dispose de caractéristiques qui déterminent sa capacité à survivre, se développer et produire dans des conditions données :

- **Besoins en eau et en lumière** : Chaque plante a un seuil optimal d'hydratation et d'ensoleillement. Un écart trop important par rapport à ces besoins peut nuire à sa croissance ou entraîner sa mort.
- **Température tolérée** : Chaque espèce supporte une plage de température spécifique au-delà de laquelle elle ne peut survivre.
- **Sol préféré** : Les plantes poussent mieux sur certains types de sol (sableux, humifère, etc.) qui leur sont naturellement adaptés.
- **Durée de vie et vitesse de croissance** : Certaines plantes poussent plus vite que d'autres, mais peuvent aussi avoir une durée de vie plus courte. D'autres ont une croissance plus lente mais offrent une production plus importante à long terme.

1.3.2. Phase de croissance

Les plantes passent par plusieurs phases de croissance : du semi à l'âge adulte. Elles ne deviennent productives qu'une fois la maturité atteinte. Le passage d'un stade à l'autre

dépend du respect des conditions environnementales : un bon arrosage, un ensoleillement adapté, un sol approprié et des températures favorables.

Lorsque les conditions optimales sont réunies, la croissance est accélérée. À l'inverse, en cas de carences ou de maladie, la croissance peut ralentir, voire la plante peut mourir.

Les plantes de grande taille nécessitent plusieurs emplacements pour se développer. Lorsqu'une plante atteint l'âge adulte, elle tente d'étendre ses racines et sa taille dans une direction (vers la droite). Si l'espace est insuffisant ou bloqué, la plante meurt. Le joueur doit donc anticiper la disposition des plantations.

1.3.3. Production et récolte

Une plante adulte génère automatiquement des productions (légumes, fruits, fleurs, etc.), que le joueur peut récolter à tout moment. Cependant, si la plante meurt ou devient malade avant la récolte, les productions sont perdues.

Certaines plantes ont une production modérée mais rapide, d'autres ont une production plus abondante mais nécessitent une gestion plus attentive.

1.3.4. Sensibilité aux maladies

Certaines maladies peuvent affecter les plantes, avec une probabilité propre à chaque espèce. Une plante malade cesse de produire et meurt au bout de deux semaines si aucun traitement n'est appliqué. Le joueur a la possibilité de soigner une plante malade à l'aide d'une action spécifique.

Chaque plante possède plusieurs types de maladies auxquelles elle est plus ou moins sensible, comme le mildiou, l'oïdium ou encore une carence en fer. Ces risques s'ajoutent aux aléas environnementaux et nécessitent une surveillance régulière.

1.4. Facteurs environnementaux

Le développement des plantes dans ENSemenC est influencé par plusieurs facteurs environnementaux simulés de manière aléatoire ou déterminée selon le monde de jeu.

1.4.1. Météo

Chaque semaine, une météo prévisionnelle est générée et affecte l'ensemble des plantes du potager. La météo combine plusieurs paramètres :

- **Température (°C)** : Si la température s'éloigne trop des tolérances d'une plante, celle-ci peut en souffrir ou mourir.
- **Ensoleillement (%)** : Un ensoleillement trop faible ou trop intense impacte la croissance.
- **Précipitations (%)** : Elles influencent le taux d'arrosage des plantes automatiquement, en complément ou en excès du soin apporté par le joueur.

Ces éléments imposent au joueur une adaptation continue dans la gestion de ses cultures.

1.5. Double temporalité du jeu

Le gameplay d'ENSemenC repose sur deux modes temporels distincts, chacun introduisant une dynamique de jeu différente.

1.5.1. Mode classique

Le mode classique est le coeur du jeu. Ce mode simule un rythme régulier de culture et de gestion. Chaque tour représente une semaine écoulée durant laquelle le joueur peut effectuer au maximum 8 actions parmi : arroser, semer, désherber, récolter, et soigner.

À la fin de chaque semaine :

- La météo prévisionnelle est appliquée.
- Les plantes évoluent selon leurs besoins, grandissent, produisent ou meurent si les conditions de survies ne sont pas réunies.
- Un évènement aléatoire peut survenir, déclenchant une urgence.

1.5.2. Mode urgence

Le mode urgence peut se déclencher aléatoirement entre deux semaines (avec une probabilité de 60 %). Il représente une situation critique à traiter en temps réel par le joueur.

Trois types d'évènements urgents peuvent survenir selon le monde (cf. Nuisibles du monde Plaine Champignon et nuisibles du monde Désert Chomp). Le joueur dispose alors de trois tentatives pour repousser la menace via différentes actions (ex. : activer un répulsif, asperger d'eau, frapper le sol...). Chaque action a une probabilité de réussite. Si toutes échouent, la plante ciblée est affectée.

Le mode urgence apporte une dimension imprévisible et dynamique, forçant le joueur à réagir rapidement et à prioriser ses décisions.

2. Déroulement d'une partie

2.1. Démarrage

Une fois le jeu lancé, le joueur est accueilli par un écran d'introduction stylisé suivi d'un menu de sélection des mondes disponibles (cf. Figure 1 et 2).



Figure 1 - Écran d'introduction d'ENSEmenC

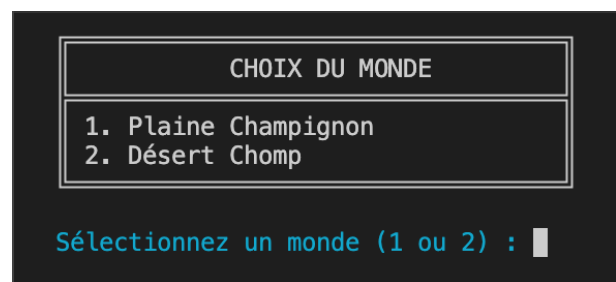


Figure 2 - Menu de sélection du monde

Le choix du monde a une incidence directe sur les mécaniques de jeu, puisqu'il détermine :

- Le type de plantes accessibles en début de partie.
- Le type de sol présent sur les terrains.
- Les conditions météorologiques rencontrées.
- Les nuisibles susceptibles d'apparaître.

Une fois le monde sélectionné, une description détaillée de celui-ci est affichée pour permettre au joueur de mieux anticiper les conditions de culture. Le joueur doit ensuite confirmer son choix. En cas d'erreur ou de doute, il peut revenir au menu pour modifier sa sélection.

Une fois le monde validé, les étapes suivantes sont automatiquement réalisées :

- **Initialisation des terrains** : deux terrains 5x5 sont mis à disposition du joueur. Ils sont associés à un type de sol compatible avec le monde choisi.

- **Initialisation de l'inventaire** : le joueur reçoit deux graines pour chaque type de plante disponible dans le monde sélectionné.
- **Affichage des consignes** : un écran récapitulatif présente les règles du jeu, les mécaniques de base, les risques environnementaux, ainsi que le fonctionnement du mode urgence.

À l'issue de cette phase préparatoire, la première semaine de culture peut débuter.

2.2. Menu d'actions et interface du jeu

Chaque semaine, le joueur peut effectuer jusqu'à 8 actions pour gérer son potager. Une action correspond à une opération effectuée sur une plante, sur un emplacement vide, ou sur l'inventaire du joueur.

Les actions disponibles sont les suivantes :

- 1) **Arroser** une plante pour ajuster son taux d'hydratation en lui ajoutant 50%.
- 2) **Récolter** les productions d'une plante adulte.
- 3) **Semer** une nouvelle graine sur un emplacement libre.
- 4) **Désherber** (retirer une plante d'un emplacement).
- 5) **Soigner** une plante atteinte d'une maladie.
- 6) **Passer au tour suivant** sans utiliser toutes ses actions.
- 7) **Quitter la partie**.

Chaque action déclenche une séquence d'interactions spécifiques : sélection du terrain, saisie des coordonnées, confirmation ou message d'erreur en cas d'action impossible.

En cas d'erreur de saisie, un message explicite s'affiche en rouge (cf. Figure 3).

```

Que faire ? (Actions restantes : 8/8)

1. Arroser
2. Récolter
3. Semer
4. Désherber
5. Soigner
6. Passer au tour suivant
7. Quitter le jeu

Entrez le numéro de l'action que vous souhaitez réaliser : jv
Choix invalide.

```

Figure 3 - Message d'erreur lors d'une action invalide

Le menu d'action est constamment ré-affiché tant que le joueur a des actions restantes.

L'interface met également à jour et affiche après chaque action (cf. Figure 4) :

- La météo prévisionnelle de la semaine.
- Les deux terrains sous forme de grilles 5x5 avec les symboles des plantes.
- L'état détaillé des plantes présentes (croissance, santé, production).
- L'inventaire de graines et de récoltes accumulées.



— Plaine Champignon — Semaine 1 —

MÉTÉO PRÉVISIONNELLE DE LE SEMAINE À VENIR

Température : 12°C - Ensoleillement : 30% - Pluie : 60%

Terrain 1

	1	2	3	4	5
1					
2					
3					
4					
5					

Terrain 2

	1	2	3	4	5
1					
2					
3					
4					
5					

MES PLANTES

Terrain 1 :
(Aucune plante plantée pour l'instant)

Terrain 2 :
(Aucune plante plantée pour l'instant)

INVENTAIRE

Graines disponibles :

1 - Tomate : 2 graine(s) restante(s)
2 - Fraise : 2 graine(s) restante(s)
3 - Champignon : 2 graine(s) restante(s)
4 - Marguerite : 2 graine(s) restante(s)

Figure 3 - Affichage de la météo, des terrains, des plantes plantées, et de l'inventaire mis à jour entre chaque action

2.3. Tour de jeu hebdomadaire (mode classique)

Chaque tour de jeu correspond à une semaine écoulée dans le potager. Le cycle hebdomadaire suit toujours le même schéma, permettant au joueur d'interagir avec ses plantations, puis d'observer les effets du temps et de la météo.

Un tour hebdomadaire se déroule ainsi :

- 1) **Début de semaine** : Une météo prévisionnelle est tirée aléatoirement parmi celles disponibles dans le monde choisi. Elle indique la température, l'ensoleillement et les précipitations prévues pour la semaine.
- 2) **Phase d'actions du joueur** : Le joueur dispose de 8 actions maximum pour gérer ses cultures. Il peut les entretenir, planter de nouvelles graines ou soigner les plantes malades.
- 3) **Passage automatique au tour suivant** : Dès que les 8 actions sont réalisées, ou si le joueur décide de passer la semaine, le jeu applique les effets de la météo et du temps sur les plantes. Ces effets incluent :
 - L'ajustement du taux d'arrosage par les précipitations.
 - La vérification de la température par rapport aux seuils de tolérance de chaque plante.
 - L'évolution de la croissance (si les conditions sont réunies).
 - Le vieillissement, l'agrandissement, et la production automatique des plantes adultes.
 - Une éventuelle contamination par des maladies.
- 4) **Passage automatique au tour suivant** : Avant de clore la semaine, le jeu évalue la probabilité de déclencher une urgence (60%). Si c'est le cas, le joueur entre dans une phase spécifique.

Ce déroulement cyclique rythme la progression du jeu, obligeant le joueur à planifier ses actions de façon stratégique tout en s'adaptant aux conditions changeantes et aléatoires.

2.4. Mode urgence

Le mode urgence est décrit par un évènement imprévisible qui introduit une situation critique nécessitant une réponse rapide du joueur. Il peut se déclencher à la fin de chaque tour avec une probabilité de 60%.

Lorsqu'un évènement urgent survient, l'affichage se modifie pour alerter le joueur. Un nuisible spécifique au monde sélectionné entre en scène pour attaquer une des plantes du potager.

Déroulement d'un mode urgence :

- 1) **Sélection aléatoire d'un nuisible** parmi ceux définis pour le monde.
- 2) **Détermination d'une cible valide** (plante vivante, productive, ou en croissance selon le nuisible).
- 3) **Affichage dynamique** de l'arrivée du nuisible sur le terrain (avec délai simulé à l'aide de *Thread.Sleep*).
- 4) **Phase de défense** :
 - Le joueur dispose de 3 actions maximum pour tenter de protéger sa plante ciblée du nuisible.
 - Trois choix lui sont proposés (ex. : frapper le sol, activer un répulsif, arroser la zone), chacun avec une probabilité de succès différente.
 - Le joueur peut aussi choisir de ne rien faire, ce qui annule les essais restants et laisse le nuisible appliquer ses effets négatifs à la plante ciblée.
- 5) **Issus de l'évènement** :
 - En cas de réussite, le nuisible prend la fuite sans causer de dégât.
 - En cas d'échec, une conséquence spécifique s'applique à la plante ciblée : perte de la plante, perte de production, ou croissance retardée.

2.5. Fin de partie

Le jeu ENSemenC ne repose pas sur une condition explicite de victoire ou de défaite. Le joueur fait évoluer son potager semaine après semaine, à son rythme, en fonction de ses objectifs personnels (récolte, entretien, expérimentation...).

La fin de partie intervient uniquement lorsque le joueur choisit de quitter le jeu, en sélectionnant l'option correspondante dans le menu des actions hebdomadaires. Un message de clôture est alors affiché pour remercier le joueur de sa participation.

3. Conception orientée objet

3.1. Structure générale du projet

La conception modulaire du code est un choix essentiel pour assurer la séparation des responsabilités. Chaque fichier ou dossier du projet remplit une responsabilité bien définie (cf. Figure 4) :

- **Program.cs** : Point d'entrée du jeu.
- **Simulation.cs** : Logique principale de la partie.
- **Affichage.cs** : Fonctions d'affichage.
- **Dossier mondes** : Définition des mondes.
- **Dossier plantes** : Définition des plantes.
- **Dossier enum** : Énumérations utilisées globalement (états, croissances, types, etc.).
- **Terrain.cs** : Gestion des plantes sur une grille.
- **Meteo.cs** : Représentation et application de la météo.

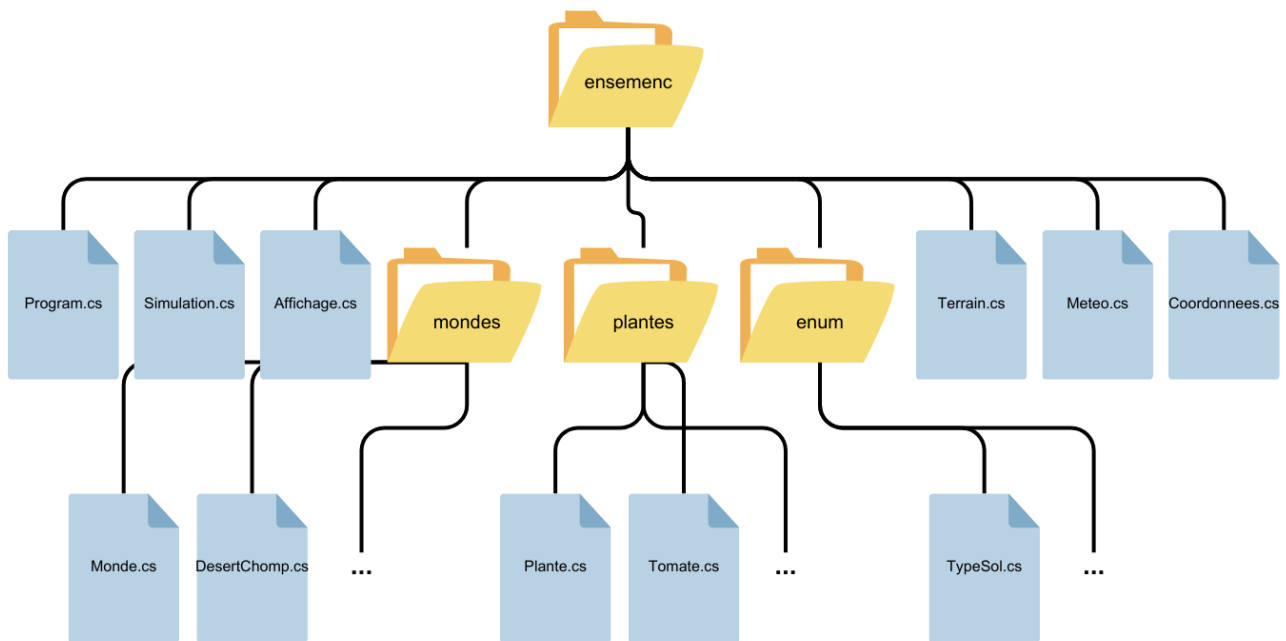


Figure 4 - Organisation des fichiers et dossiers du jeu ENSemenc

3.2. Hiérarchie des classes

3.2.1. Mondes

Le jeu propose plusieurs mondes, chacun possédant ses propres règles de culture, types de plantes et nuisibles. Pour représenter cela, une classe abstraite *Monde* a été définie. Elle contient :

- Le nom du monde.
- La météo prévisionnelle de la semaine.
- La liste des météos possibles.
- Les types de sols disponibles dans ce monde.
- Les types de plantes cultivables.
- Une illustration ASCII du monde.
- Les méthodes de gestion des urgences.
- Des méthodes abstraites à implémenter dans les classes filles, comme *AfficherDescription()* ou *LancerUrgence()*.

Deux mondes héritent de la classe *Monde* :

- PlaineChampignon : monde tempéré, équilibré pour débiter.
- DesertChomp : monde aride, plus difficile à gérer.

Cette structure permet d'ajouter facilement de nouveaux mondes, en définissant simplement leurs spécificités dans une nouvelle classe fille de *Monde*. La logique d'urgence, par exemple, est polymorphe : chaque monde gère ses propres types de nuisibles via l'override de *LancerUrgence()*.

3.2.2. Plantes

La classe abstraite *Plante* regroupe l'ensemble des attributs et comportements communs aux espèces :

- Caractéristiques biologiques (besoins, température, sol, espérance de vie...).
- État de santé et phase de croissance.
- Mécanismes de production, vieillissement, contamination.
- Comportements liés à l'environnement (croissance accélérée ou mort selon conditions météo).
- Gestion de la taille et de l'occupation sur le terrain (plantes étendues sur plusieurs cases)

Chaque espèce (ex. : Tomate, Cactus) hérite de *Plante* et définit uniquement ses caractéristiques spécifiques et ses sensibilités aux maladies depuis le constructeur.

Ce choix permet une spécialisation par plante tout en mutualisant la logique. Le modèle est extensible : pour ajouter une nouvelle plante, il suffit d'hériter de *Plante* et de définir ses valeurs dans le constructeur. Cette structure permet également de développer des spécificités selon les plantes dans des versions supplémentaires du jeu.

3.2.3. Terrain

La classe *Terrain* représente une grille de 5 par 5 cases où les plantes sont placées. Elle gère l'ensemble des actions que le joueur peut effectuer pour ses cultures :

- Semer une graine.
- Arroser une parcelle.
- Récolter des production.
- Désherber une parcelle (supprimer une plante).
- Soigner une plante malade.
- Mettre à jour toutes les plantes selon les conditions environnementales.

Chaque terrain connaît son type de sol et maintient une liste des plantes vivantes ainsi qu'un tableau deux dimensions des emplacements représentant les parcelles. Le terrain permet de faire le lien entre l'environnement, les actions du joueur, et l'évolution des plantes.

3.2.4. Coordonnées

La classe *Coordonnees* est une simple classe permettant de stocker les positions x et y d'une plante sur la grille du terrain. Elle est utilisée dans les interactions du joueur (sélection d'un emplacement), la croissance des plantes (extension à droite, en ligne), et les attaques des nuisibles (affichage des emplacements visés).

En encapsulant les coordonnées dans une classe dédiée, on évite les erreurs de manipulation directe de paire (*int*, *int*) et on rend le code plus lisible et évolutif.

3.2.5. Météo

La classe *Meteo* représente les conditions environnementales d'une semaine de jeu. Chaque météo possède trois paramètres : température en degrés, taux d'ensoleillement en pourcentage, et taux de précipitation en pourcentage également.

Chaque monde dispose d'une liste de météos possibles, dans laquelle une météo est tirée au hasard à chaque tour. Les plantes adaptent ensuite leur comportement (croissance, maladie, mort...) en fonction de cette météo.

3.2.6. Simulation

La classe *Simulation* est le moteur principal du jeu. Elle gère l'ensemble du cycle de vie d'une partie :

- Initialisation du monde, des terrains et de l'inventaire du joueur.
- Affichage des menus et consignes.
- Boucle principale du jeu : actions du joueur, mise à jour des plantes, déclenchement d'urgences.
- Appels à toutes les autres classes selon les interactions du joueur.

En centralisant la logique de la partie dans *Simulation*, on sépare clairement la logique métier (plantes, monde, météo...) du contrôle de jeu (tour par tour, gestion des entrées, affichage).

3.2.7. Affichage

La classe *Affichage* regroupe toutes les fonctions d'interface console : introduction, menus, affichage des terrains, météo, inventaire, erreurs, confirmations, etc. Toutes les

sorties utilisateur passent par cette classe, ce qui facilite l'homogénéité de l'affichage et les évolutions futures (ajout de style, transition vers une interface graphique, etc.).

3.3. Diagramme UML

Le diagramme de classes UML global du projet ENSemenC est présenté en annexe de ce rapport. Il permet de visualiser la structure générale du code, les relations d'héritage entre les entités, ainsi que les principales associations entre les classes (monde, plante, terrain, simulation, etc.).

Ce diagramme a été réalisé à l'aide de PlantUML, un outil de modélisation textuel que j'ai eu l'occasion d'utiliser dans le cadre de précédents projets. Il permet de maintenir une bonne lisibilité et de produire facilement des visuels.

4. Tests et vérifications

4.1. Stratégie de test adoptée

Les tests du projet ENSemenC ont été réalisés de manière manuelle tout au long du développement. Chaque fonctionnalité a été testée individuellement après son implémentation, puis validée à l'aide de scénarios utilisateur reproduits via l'interface console.

Le développement ayant été mené de manière itérative, les tests ont suivi une logique incrémentale : après chaque ajout ou modification, des vérifications ont été effectuées pour s'assurer du bon fonctionnement du nouveau code, ainsi que de l'absence de régression sur les fonctionnalités précédemment intégrées.

Grâce à une architecture claire et modulaire, la détection et la correction des éventuels dysfonctionnements ont été facilitées.

4.2. Vérification des fonctionnalités principales

Les tests ont permis de valider le bon fonctionnement des actions principales proposées au joueur :

- **Semis** de différentes plantes selon les règles du monde sélectionné.
- **Arrosage** influençant correctement le taux d'hydratation et la survie des plantes.
- **Récolte** de productions.
- **Désherbage** avec gestion des plantes composées (plante parente + filles).
- **Soins** des plantes malades selon leur état de santé.
- **Déroulement des urgences** avec sélection aléatoire de nuisibles, affichage correct, gestion des tentatives et effets associés.

Chaque action a été vérifiée sur différents cas (plantes saines, malades, mortes, emplacements vides, sol non adapté...).

4.3. Cas particuliers testés

Des cas limites ont été volontairement simulés pour garantir la robustesse de l'application :

- Semer sur un emplacement déjà occupé.
- Arroser ou soigner une plante morte.
- Tenter une action semer, arroser, désherber, ou soigner, sur une parcelle de terrain vide.
- Récolter une plante sans production.
- Gérer une plante atteignant le bord du terrain sans pouvoir grandir.
- Lancer un mode urgence sans plante disponible à attaquer.

Ces tests ont permis d'identifier quelques ajustements à effectuer, notamment pour affiner certains messages d'erreur ou sécuriser des appels. De plus, tous ces cas ne produiront pas d'erreur, mais afficherons un message d'avertissement à l'utilisateur pour lui signifier que son action n'a pas eu d'effet.

5. Gestion de projet

5.1. Contexte du projet

Le développement s'est effectué seule, ce qui a nécessité une rigueur particulière dans l'organisation du code, la documentation et la planification des différentes étapes. Le cahier des charges imposait un ensemble de fonctionnalités minimales (cultures, météo, actions hebdomadaires, interactions avec l'environnement), tout en laissant une grande liberté dans le choix de l'univers ou des mécaniques complémentaires.

5.2. Inspirations créatives

Pour rendre l'univers du jeu plus original et immersif, l'univers d'ENSemenc s'inspire directement de l'univers Mario, en adaptant ses codes visuels et ludiques au contexte de la simulation agricole. L'idée de structurer le jeu en mondes thématiques (Plaine Champignon, Désert Chomp, etc.) permet non seulement de diversifier les mécaniques de jeu, mais aussi de renforcer la cohérence interne. Chaque monde dispose de ses propres plantes, nuisibles, spécificités météorologiques et éléments visuels.

À l'origine, cinq mondes avaient été envisagés, chacun doté d'un ensemble de règles et mécaniques spécifiques. Seuls deux ont été intégrés à cette version du jeu (Plaine

Champignon et Désert Chomp), mais les autres descriptions ont été conservées pour d'éventuelles extensions et sont disponibles en annexe.

5.3. Organisation

Le développement a suivi une approche progressive et modulaire, facilitée par une phase de conception approfondie en amont. Chaque nouvelle fonctionnalité a fait l'objet d'une réflexion sur les interactions avec les éléments déjà en place, et a été testée manuellement avant validation.

Phase	Objectif	Détails
1. Analyse et conception	Comprendre le sujet, imaginer l'univers du jeu, poser les bases de la structure objet.	Lecture attentive du sujet, choix de l'univers Mario comme fil conducteur, description des mondes, réalisation d'un premier diagramme UML sur PlantUML, description d'une partie de jeu simple et complète.
2. Développement d'une version minimale	Créer une base fonctionnelle permettant de lancer une partie.	Lancement d'une partie, actions de base (semer, arroser, etc.), mise à jour des plantes hebdomadaires.
3. Ajout des fonctionnalités avancées	Enrichir l'expérience de jeu.	Intégration des maladies (état, durée, traitement), mise en place du mode urgence avec nuisibles et actions spécifiques, ajout d'un monde supplémentaire, amélioration de l'interface console, gestion de la croissance.

Phase	Objectif	Détails
5. Finalisation	Stabiliser le code, vérifier le bon fonctionnement global.	Documentation du code pour en faciliter la lecture (réalisée tout au long du projet, mais ajustée à la fin), README, mise à jour du diagramme UML, tests supplémentaires, rédaction du rapport technique.

Remarques sur les priorités :

- Le mode classique a été priorisé pour garantir un jeu fonctionnel rapidement.
- Les fonctions d'affichage ont été soignées dès le début pour améliorer l'expérience utilisateur (couleurs, symboles, menus).
- L'ajout des éléments de gameplay avancé s'est fait uniquement une fois la stabilité de base assurée.
- Une attention particulière a été portée à la documentation du code, pour faciliter sa compréhension et sa maintenance (et sa correction).

6. Bilan critique

Le développement du projet ENSemenC a représenté un véritable défi, notamment du fait qu'il a été entièrement mené en autonomie. Si les compétences techniques en programmation étaient suffisantes pour mener à bien le projet, la complexité principale résidait clairement dans la phase de conception.

Le sujet, ouvert et exigeant, laissait une large liberté d'interprétation tout en imposant de nombreux mécanismes à implémenter (gestion d'un potager, influence de la météo, croissance de plantes, aléas urgents, etc.). Sans une modélisation rigoureuse, il était facile de s'éparpiller ou de rencontrer des blocages en phase de développement. C'est

pourquoi une phase d'analyse et de conception structurée a été privilégiée, ce qui a permis de gagner du temps par la suite lors du codage.

Le fait d'être seule à travailler sur le projet a renforcé la difficulté. Concevoir un projet seul limite les possibilités de confrontation d'idées et de validation mutuelle des choix de conception. Un binôme aurait sans doute facilité cette phase, ne serait-ce que pour discuter de l'architecture ou partager des hypothèses de modélisation.

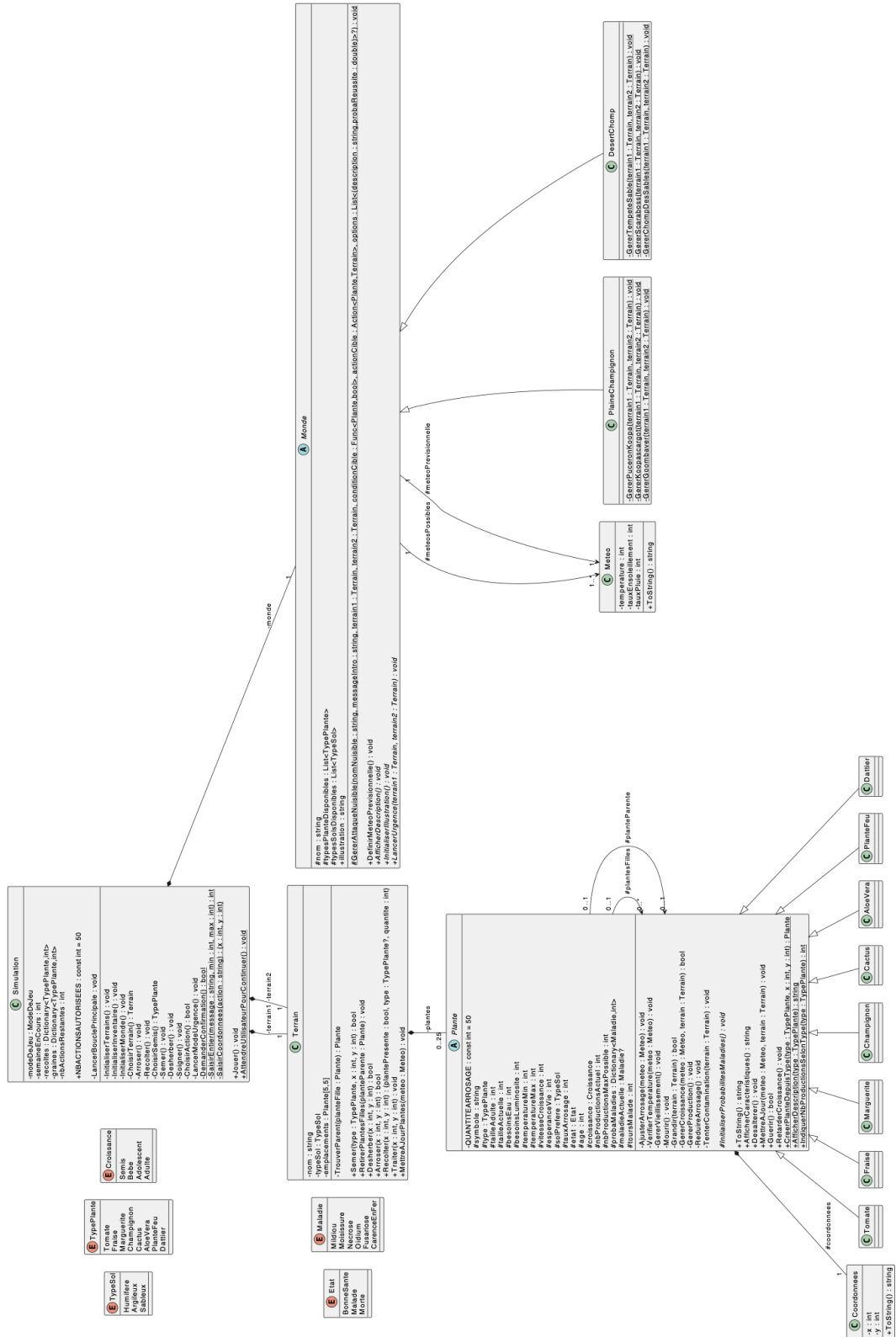
Malgré cela, le projet livré est complet, stable, fonctionnel et bien structuré. Il respecte l'ensemble des critères attendus dans le sujet, tout en proposant une couche d'originalité, notamment à travers l'univers inspiré de Mario et la description de plusieurs mondes fictifs. Bien que seule une partie de ces mondes ait été effectivement implémentée (Plaine Champignon et Désert Chomp), les idées complémentaires ont été documentées et pourraient faire l'objet d'un développement futur.

Compte tenu de la charge de travail importante à cette période de l'année (autres projets, partiels, oraux...), le choix a été fait de prioriser la qualité sur la quantité. L'objectif était de livrer une version soignée, documentée et maintenable, plutôt que de viser une exhaustivité risquée au détriment de la clarté et de la stabilité.

En résumé, ENSemenC a été une expérience enrichissante tant sur le plan technique que méthodologique. Il a permis de mobiliser de nombreuses compétences (programmation objet, conception UML, architecture logicielle, tests manuels) et a mis en évidence l'importance cruciale d'une modélisation solide en amont, en particulier pour les projets orientés objets à forte complexité fonctionnelle.

Annexes

Annexe 1 : Diagramme UML général des classes du projet



Annexe 2 : Description initiale des 5 mondes du jeu

Modélisation ENSemenc

1. Description des mondes (inspiration Mario)

Monde 1 - Plaine champignon

Le monde d'introduction classique, inspiré des plaines de Mario bros.

Description : Bienvenue dans la Plaine Champignon, un monde doux et paisible où tout pousse...
sauf quand un Goombavers passe par là !

- Température : tempéré (10 à 25°C)
- Météo : alternance plus/soleil, météo équilibrée, semblable au printemps
- Plantes : Tomates, Fraises, Marguerites, Champignons
- Nuisibles : Goombavers (vers de terre glouton), Koopascargots (escargots) ou Pucerons koopa (pucerons)
- Spécificités :
 - Les champignons peuvent muter (champignons classique peut devenir un champignon rare de type shiitake, ou vénéneux).
 - Apparition de Toad le jardinier qui peut booster la croissance des plantes.

Monde 2 - Désert chomp

Inspiré du désert Mario, avec chaleur extrême, tempêtes de sable, cactus et plantes résistantes.

Description : Le Désert Chomp brûle sous un soleil de plomb. Ici, seules les plantes les plus résistantes survivent aux crocs du sable...

- Température : élevée (30 à 45°C)
- Météo : très sec, rares tempêtes de sable
- Plantes : Cactus, Aloé vera, Plante feu, Dattier

- Nuisibles : Chomps des sables (creuse et détruit les racines), Scaraboss (scarabés), une tempête de sable peut balayer et éliminer les plantes peu résistante (niveau de vie faible).
- Spécificités :
 - Les plantes feu peuvent repousser les nuisibles si elles sont bien cultivées, mais peuvent détruire les plantes alentours si elles sont trop proches.

Monde 3 - Jungle wiggler

Inspiré de la jungle de Mario et de la forêt amazonienne, pleine de vies inconnues et de dangers.

Description : Une jungle dense et étouffante où la végétation déborde de toute part. La croissance est rapide... tout comme les dangers qui surgissent entre les lianes.

- Température : chaude et humide (25 à 35°C)
- Météo : pluie fréquente, chaleur constante, taux d'humidité très élevé
- Plantes : Plante carnivore, Liane magique (plante grimpante), Plante hallucinogène, Bananier
- Nuisibles : Wiggler (mange la tige), Moustikooa (peut empêcher des actions)
- Spécificités :
 - Croissance très rapide.
 - Risque élevé de maladies.
 - Les plantes hallucinogènes peuvent empêcher certaines actions (ou en obliger...).

Monde 4 - Royaume sorbet

Le monde glacial inspiré du circuit Mario Kart du même nom, où seules les plantes robustes survivent.

Description : Le Royaume Sorbet est une étendue glacée où le moindre souffle est givré... et où les plantes bravent la neige !

- Température : froide (-15 à 5°C)
- Météo : chute de neige et blizzards
- Plantes : Navets, Topinambours, Sapin décoratif, Fleurs de glace
- Nuisibles : Pingouin bandit (vole les productions), Topi taupe (creuse des trous dans le jardin)

- Spécificités :
 - Les fleurs de glace peuvent ralentir les nuisibles.
 - Besoin de pare-vent.
 - Les plantes gèlent en cas d'arrosage excessif.

Monde 5 - Tropique Lakitu

Le monde coloré, humide, inspiré des plages et jungles exotiques de Mario.

Description : Sous le soleil du Tropique Lakitu, tout est rythme, fruits mûrs... et attaques surprises depuis les nuages !

- Température : douce et stable (20 à 30°C)
- Météo : ensoleillé et orages fréquents
- Plantes : Ananas, Mangue, Hibiscus, Plante tambour (plante musicale)
- Nuisibles : Lakitu orageux (lance des éclairs), Crabes (creuse)
- Spécificités :
 - Les orages peuvent détruire des plantes ou faire fuir des crabes.
 - La plante tambour booster les plantes voisines.