

**Hadrien Beaurepaire**  
**Ghali Benani**  
**Nael Belhadj Amara**  
**Julia Guimaraes**  
**Melissa Gay**

# CDSF

## WHAT'S NEXT?

Nom du projet: What's next?

Version 1: 10/01/2026

### Équipe:

### Présentation du groupe:

Afin d'assurer la bonne réalisation et le déroulement optimal de notre projet de jeu vidéo, nous avons mis en place une organisation claire, structurée et méthodique. Cette organisation repose sur une répartition précise des rôles et des responsabilités, que nous considérons comme un élément essentiel à la réussite du projet, tant du point de vue du respect des délais que de la qualité finale du produit.

Nous avons fait le choix d'une répartition volontairement restreinte mais fortement thématique des tâches, dans le but de limiter les chevauchements de responsabilités, de favoriser la spécialisation de chaque membre de l'équipe et d'optimiser l'efficacité du travail collaboratif. Le projet est ainsi structuré autour de six pôles fonctionnels distincts, présentés ci-dessous.

### Répartition des pôles

#### 1. Programmation

Le pôle programmation constitue la base technique du projet. Il est chargé de l'implémentation de l'ensemble des fonctionnalités définies par les autres pôles, en traduisant les concepts et mécaniques de jeu en code opérationnel. Ce pôle inclut notamment le développement des systèmes de jeu, la gestion des interactions, l'implémentation des mécaniques principales ainsi que l'optimisation technique et la maintenance du code. Le responsable de ce pôle est Ghali, qui assure la coordination du développement technique et veille à la stabilité, à la cohérence et à la qualité du code tout au long du projet.

#### 2. Level Design

Le pôle level design est un élément central de l'expérience de jeu, dans la mesure où il structure la progression du joueur et les défis qui lui sont proposés. Il est chargé de la conception et de l'agencement des différents niveaux du jeu, en tenant compte de la progression, de l'équilibrage, du rythme et de la cohérence globale avec les mécaniques de gameplay.

Ce pôle est placé sous la responsabilité de Melissa, qui veille à proposer des niveaux cohérents, équilibrés et adaptés à l'expérience de jeu souhaitée.

### **3. Game Design**

Le pôle game design regroupe l'ensemble des éléments conceptuels définissant le fonctionnement et l'identité du jeu. Il comprend la définition des règles, des mécaniques de gameplay, des interactions, ainsi que l'intégration d'éléments visuels et ergonomiques facilitant la compréhension du jeu par le joueur et renforçant son attractivité.

Le responsable de ce pôle est Nael, garant de la cohérence globale du gameplay et de la qualité de l'expérience de jeu proposée.

### **4. Intelligence artificielle et réseaux**

Ce pôle est chargé de la conception et de l'implémentation de l'intelligence artificielle du jeu, notamment le comportement des entités non-joueurs, ainsi que du développement des fonctionnalités liées au multijoueur et aux communications réseau.

Le responsable de ce pôle est Hadrien, qui veille au bon fonctionnement des systèmes d'intelligence artificielle et des interactions réseau, tout en assurant leur intégration cohérente avec les autres composantes du projet.

### **5. Audio**

Le pôle audio est responsable de la création et de l'intégration de l'environnement sonore du jeu. Il inclut la conception des musiques, des effets sonores et des éléments audio interactifs contribuant à l'immersion du joueur.

Ce pôle est également placé sous la responsabilité de Hadrien, qui veille à la cohérence artistique et à la qualité sonore du projet.

### **6. Organisation**

Le pôle organisation assure la gestion globale du projet. Il est chargé de la planification des différentes phases de développement, du suivi de l'avancement, du respect des délais, de l'organisation des réunions et de la coordination entre les différents pôles.

La responsable de ce pôle est Julia, garante du bon fonctionnement de l'équipe et de la communication interne.

## **Outils de communication et de développement**

Afin d'assurer une collaboration efficace et une communication centralisée, l'équipe utilise un serveur Discord dédié, organisé en canaux thématiques correspondant aux différents pôles du projet. Cette structure permet de séparer les échanges par domaine, de faciliter le suivi des discussions et d'améliorer la coordination entre les membres de l'équipe.

Le développement technique du projet est réalisé à l'aide de Visual Studio Code comme environnement de développement principal, associé à Git pour la gestion de versions. L'utilisation de Git permet un travail collaboratif structuré, le suivi des modifications, la prévention des conflits et la sécurisation du code source.

Enfin, la documentation du projet est centralisée à travers des documents partagés hébergés sur une plateforme cloud, garantissant un accès permanent aux ressources du projet, une mise à jour continue des informations et une meilleure traçabilité des décisions prises.

## Contexte et objectifs:

### Présentation succincte du besoin initial

Ce projet s'inscrit dans le cadre de la SAE J3D de première année à l'EPITA, dont la consigne est de concevoir et de développer, en groupe de quatre à cinq étudiants, un jeu vidéo tout au long de l'année universitaire. Ce projet informatique constitue un travail de longue durée visant à mettre en application les connaissances acquises en cours et en travaux pratiques de programmation, tout en mobilisant des compétences complémentaires nécessaires à la réalisation d'un projet concret.

Conformément à la consigne, le jeu doit être développé en Python, utiliser le framework graphique PyGame, intégrer une forme d'intelligence artificielle simple ainsi qu'un mode multijoueur en réseau. Le projet doit également être fonctionnel sur Windows 10/11 et respecter les contraintes techniques et organisationnelles définies par l'encadrement pédagogique.

La première étape de ce projet consiste en la rédaction d'un cahier des spécifications techniques, destiné à définir précisément le périmètre, les objectifs et les choix de conception du jeu avant le début du développement.

### Problématique

La problématique centrale du projet est la suivante :  
Comment concevoir et développer un jeu vidéo original répondant aux contraintes pédagogiques, techniques et temporelles imposées par la SAE J3D, tout en proposant une expérience de jeu cohérente, immersive et engageante pour les joueurs ?

Cette problématique implique de concilier les exigences académiques (langage, framework, fonctionnalités obligatoires) avec des choix de game design pertinents, une organisation efficace du travail en équipe et une architecture logicielle adaptée, notamment pour la gestion du réseau et de l'intelligence artificielle.

## Objectifs du projet

### Objectifs pédagogiques :

- Mettre en pratique les connaissances acquises en programmation au cours de la première année à l'EPITA
- S'initier au développement d'un projet informatique de longue durée
- Apprendre à travailler en équipe et à répartir les rôles et responsabilités
- Découvrir les enjeux de la gestion de projet, de la communication et de la coordination
- Se confronter à des contraintes techniques réelles, notamment liées au réseau et à l'architecture logicielle

### Objectifs fonctionnels :

- Concevoir un jeu vidéo coopératif mêlant énigmes et suspense
- Implémenter un mode multijoueur en réseau conforme aux exigences de la SAE
- Intégrer une intelligence artificielle simple influençant le gameplay
- Proposer une expérience de jeu basée sur la coopération, la prise de décision et des choix à conséquences
- Livrer un produit fonctionnel, stable et conforme aux spécifications définies dans le cahier des charges

## Périmètre du projet:

### Ce que le projet couvre :

- Le développement d'un jeu vidéo jouable sur Windows 10/11
- La mise en place d'un mode multijoueur en réseau, permettant à plusieurs joueurs de coopérer à distance
- L'intégration d'une intelligence artificielle simple, influençant le comportement d'entités non-joueurs ou l'environnement du jeu
- La conception de plusieurs salles ou niveaux, dans lesquels les joueurs doivent résoudre des énigmes afin de progresser
- L'implémentation de mécaniques de gameplay basées sur la coopération, la communication et la prise de décisions
- L'intégration d'éléments narratifs et de choix ayant un impact sur le déroulement et l'issue de la partie
- La création d'une ambiance visuelle et sonore contribuant au suspense et à l'immersion du joueur
- La production des livrables demandés dans le cadre de la SAE (cahier des spécifications techniques, démonstrations, soutenances, version finale du jeu)

### Ce que le projet ne couvre pas :

- Le support officiel de plateformes autres que Windows 10/11
- L'utilisation de moteurs de jeu ou de frameworks non validés par l'encadrement pédagogique

- L'intégration de fonctionnalités commerciales telles que microtransactions ou contenus payants
- Toute stratégie de communication, de marketing ou de publicité autour du jeu
- 
- Tout objectif lié à l'acquisition d'un nombre d'utilisateurs, à la diffusion à grande échelle ou à la commercialisation du produit
- 
- La monétisation du jeu (microtransactions, publicité intégrée, ventes)
- 
- Le développement d'un produit destiné à un usage professionnel ou industriel

## Parties prenantes et utilisateurs cibles :

### Utilisateurs types:

Les utilisateurs cibles de WHAT'S NEXT? sont principalement des joueurs et joueuses appréciant les expériences coopératives riches en réflexion et en narration. Le jeu s'adresse à un public adolescent et jeune adulte, habitué aux jeux vidéo mais ne cherchant pas nécessairement une grande exigence technique ou des réflexes précis. Il vise avant tout des joueurs qui aiment communiquer, collaborer et prendre des décisions ensemble, que ce soit entre amis, en couple ou entre membres d'une même famille. Sensibles aux univers immersifs, aux histoires à dimension morale et aux mécaniques de gameplay asymétriques, ces utilisateurs trouveront dans WHAT'S NEXT? une expérience originale qui valorise l'entraide, la coordination et les choix partagés, tout en proposant une réflexion plus profonde sur la dépendance, la confiance et les conséquences de nos actions.

### Autres acteurs concernés :

En plus des utilisateurs finaux, plusieurs acteurs interviennent ou sont concernés par le projet :

- **Équipe de développement:** composée des membres du groupe projet, est responsable de la conception, du développement, des tests et de la livraison du jeu. Chaque membre possède un rôle défini contribuant au bon déroulement du projet.
- **Encadrants pédagogiques de l'EPITA:** les responsables locaux et nationaux de la SAE J3D, assurent le suivi du projet. Ils valident le sujet, le cahier des spécifications techniques, évaluent l'avancement lors des soutenances et formulent des recommandations ou imposent des ajustements afin de garantir la conformité du projet aux exigences pédagogiques.
- **Jurys de soutenance:** composés d'enseignants, évaluent la qualité technique, organisationnelle et fonctionnelle du projet lors des différentes soutenances prévues au cours de l'année.
- **Testeurs:** pouvant être des membres de l'équipe ou des personnes externes au groupe, participent à l'évaluation du jeu en phase de développement afin d'identifier les bugs, problèmes d'équilibrage ou de jouabilité.

# Description générale du produit attendu:

## Vision globale du fonctionnement:

Le produit attendu est un jeu vidéo coopératif en réseau, développé en Python avec le framework PyGame, dans lequel deux joueurs incarnent des personnages évoluant dans un environnement clos composé de plusieurs salles successives.

Les joueurs doivent coopérer afin de résoudre des énigmes et surmonter différents obstacles leur permettant de progresser de salle en salle. La communication et la coordination entre les joueurs sont essentielles à la réussite, certaines mécaniques nécessitant des actions synchronisées ou des prises de décision communes.

Le jeu intègre une intelligence artificielle simple, prenant la forme d'entités ou de mécanismes réagissant aux actions des joueurs et contribuant à la difficulté, au suspense et à la dynamique du gameplay.

## Cas d'usage principaux ou scénarios typiques:

### Démarrage d'une partie :

Les joueurs lancent le jeu, se connectent via le mode multijoueur en réseau et accèdent à une partie coopérative.

### Phase de jeu principale :

Les joueurs explorent une salle, analysent l'environnement, communiquent entre eux et collaborent pour résoudre les énigmes proposées. L'intelligence artificielle peut intervenir pour modifier le comportement de l'environnement ou introduire de nouveaux défis.

### Fin de partie :

Selon les actions et décisions prises, la partie se conclut par différentes issues possibles, favorisant la rejouabilité.

## Contraintes globales:

- Développement en Python, avec l'utilisation du framework PyGame
- Fonctionnement garanti sur Windows 10/11
- Intégration obligatoire d'un mode multijoueur en réseau
- Présence d'une intelligence artificielle simple
- Respect des délais et des livrables imposés par la SAE J3D
- Ergonomie permettant une prise en main intuitive du jeu
- Stabilité et jouabilité suffisantes pour une démonstration lors des soutenances

# **Exigences fonctionnelles:**

## **Liste des fonctions attendues, organisées par thèmes:**

### **Gestion du jeu et des parties:**

#### **1 – Lancement du jeu**

- Description : Le jeu doit pouvoir être lancé par un joueur depuis un exécutable Python.
- Condition d'entrée : Le jeu est correctement installé sur un système Windows 10/11.
- Condition de sortie : Le menu principal du jeu est affiché.
- Priorité : Obligatoire

#### **2 – Création et démarrage d'une partie**

- Description : Le jeu doit permettre la création et le démarrage d'une partie coopérative.
- Condition d'entrée : Les joueurs sont connectés au réseau.
- Condition de sortie : La partie commence et les joueurs sont placés dans la première salle.
- Priorité : Obligatoire

### **Multijoueur en réseau:**

#### **1 – Connexion des joueurs en réseau**

- Description : Le jeu doit permettre à plusieurs joueurs de se connecter entre eux via un réseau afin de jouer en coopération.
- Condition d'entrée : Les joueurs disposent d'une connexion réseau fonctionnelle.
- Condition de sortie : Les joueurs sont connectés à la même partie.
- Priorité : Obligatoire

#### **2 – Synchronisation des actions**

- Description : Les actions des joueurs (déplacements, interactions) doivent être synchronisées entre les différentes instances du jeu.
- Condition d'entrée : Une partie multijoueur est en cours.
- Condition de sortie : Les actions sont visibles de manière cohérente par tous les joueurs.
- Priorité : Obligatoire

### **Gameplay et énigmes:**

#### **1 – Déplacement des personnages**

- Description : Chaque joueur doit pouvoir contrôler son personnage et se déplacer dans l'environnement de jeu.
- Condition d'entrée : Une partie est en cours.
- Condition de sortie : Le personnage se déplace selon les commandes du joueur.
- Priorité : Obligatoire

## 2 – Résolution dénigmes coopératives

- Description : Le jeu doit proposer des énigmes nécessitant la coopération des joueurs pour être résolues.
- Condition d'entrée : Les joueurs interagissent avec les éléments de la salle.
- Condition de sortie : Lénigme est résolue et permet la progression.
- Priorité : Obligatoire (Pour la cohérence du jeu )

## Intelligence artificielle:

- Description : Le jeu doit intégrer une intelligence artificielle simple réagissant aux actions des joueurs (ennemi, mécanisme, surveillance, etc.).
- Condition d'entrée : Une entité contrôlée par lIA est active dans la salle.
- Condition de sortie : Lentité adopte un comportement cohérent selon la situation.
- Priorité : Obligatoire

## Narration et choix du joueur:

### 1 – Choix influençant la progression

- Description : Le jeu doit proposer des choix ayant un impact sur le déroulement ou lissue de la partie.
- Condition d'entrée : Le joueur atteint un point clé du scénario.
- Condition de sortie : La progression du jeu est modifiée selon le choix effectué.
- Priorité : Souhaitable

### 2 – Fins multiples

- Description : Le jeu peut proposer plusieurs fins selon les décisions prises par les joueurs.
- Condition d'entrée : La partie arrive à son terme.
- Condition de sortie : Une fin spécifique est affichée.
- Priorité : Optionnelle

## Interface et retours utilisateur:

### 1 – Interface utilisateur

- Description : Le jeu doit proposer une interface claire permettant au joueur de comprendre son environnement et ses actions possibles.
- Condition d'entrée : Le jeu est en cours d'exécution.
- Condition de sortie : Les informations essentielles sont visibles et compréhensibles.
- Priorité : Obligatoire

### 2 – Retours visuels et sonores

- Description : Les actions importantes du joueur doivent être accompagnées de retours visuels et/ou sonores.
- Condition d'entrée : Une action significative est effectuée.
- Condition de sortie : Un retour est perçu par le joueur.
- Priorité : Souhaitable

## **Performances:**

### **1 – Temps de réponse**

Le jeu doit garantir un temps de réponse suffisant pour assurer une expérience fluide, sans latence perceptible lors des actions courantes (déplacements, interactions).

### **2 – Affichage fluide**

Le jeu doit maintenir une fluidité stable lors de l'affichage des scènes, compatible avec les capacités moyennes des machines utilisées par les étudiants.

## **Fiabilité et stabilité:**

### **1 – Stabilité du jeu**

Le jeu ne doit pas se fermer de manière inattendue lors d'une utilisation normale. Les erreurs critiques doivent être limitées autant que possible.

### **2 – Gestion des erreurs réseau**

Le jeu doit gérer les erreurs réseau de manière contrôlée (déconnexion, perte de connexion), sans provoquer de crash du programme.

## **Sécurité:**

Les échanges réseau doivent être limités aux données nécessaires au fonctionnement du jeu. Aucune donnée personnelle des joueurs ne doit être collectée ou stockée.

## **Maintenabilité et évolutivité:**

### **1 – Lisibilité et organisation du code**

Le code doit être structuré, commenté et organisé de manière à faciliter sa compréhension et sa maintenance par les membres de l'équipe.

### **2 – Gestion de versions**

Le projet doit utiliser un système de gestion de versions (Git) afin d'assurer le suivi des modifications et la collaboration entre les membres de l'équipe.

## **Ergonomie et accessibilité:**

### **1 – Prise en main du jeu**

Le jeu doit être compréhensible rapidement par un nouveau joueur, sans nécessiter de connaissances techniques particulières.

## 2 – Interface utilisateur

L'interface doit être claire, lisible et cohérente avec l'ambiance du jeu.

### Contraintes réglementaires et pédagogiques:

#### 1 – Respect des contraintes pédagogiques

Le jeu doit respecter l'ensemble des contraintes imposées par la SAE J3D, notamment l'utilisation de Python, PyGame, un mode multijoueur en réseau et une intelligence artificielle simple.

#### 2 – Respect des délais

Le développement du projet doit respecter le planning et les échéances fixés dans le cadre de la SAE.

## Hypothèses et dépendances:

### Hypothèses:

L'équipe part du principe que l'ensemble des membres restera impliqué jusqu'à la fin du projet. Toutefois la finalisation du projet est possible même en cas de réduction de l'effectif.

Il est supposé que les membres chargés de la programmation s'impliquent activement dans un apprentissage autonome complémentaire, afin d'acquérir les connaissances techniques nécessaires au développement du jeu.

Enfin, l'équipe part du principe que les outils utilisés tout au long du projet (framework PyGame, système de gestion de versions Git, environnement de développement Visual Studio Code et plateforme de communication Discord) fonctionnent correctement et resteront disponibles pendant toute la durée du projet.

### Dépendances:

- validation du sujet et du cahier des spécifications techniques par les encadrants pédagogiques, condition indispensable au lancement et à la poursuite du développement.
- bon fonctionnement de bibliothèques et outils externes, notamment PyGame et les modules réseau utilisés pour l'implémentation du mode multijoueur.

## Critères d'acceptation et de validation:

La validation du projet repose principalement sur les trois soutenances obligatoires prévues dans le cadre de la SAE J3D, telles que définies dans la consigne officielle.

Le projet sera considéré comme conforme et validé si les critères suivants sont respectés :

- Le cahier des spécifications techniques est validé par les encadrants pédagogiques avant le début du développement
- Le projet est présenté et évalué lors des trois soutenances prévues (première soutenance, deuxième soutenance et soutenance finale)
- Une progression concrète et mesurable du projet est démontrée entre chaque soutenance
- Le jeu présenté lors de la soutenance finale est fonctionnel, jouable et conforme aux spécifications validées
- Les fonctionnalités obligatoires imposées par la SAE (utilisation de PyGame, intelligence artificielle simple, multijoueur en réseau) sont effectivement implémentées
- Les livrables demandés (rapports, démonstrations, version finale du projet) sont fournis dans les délais impartis

La conformité du projet est évaluée par les jurys de soutenance, selon les critères définis dans la grille d'évaluation fournie par l'EPITA.

## Planning et estimation:

### Planning du projet / Jalons et avancement prévisionnel:

#### Phase 1 – Conception et lancement du projet (jusqu'à la première soutenance)

Cette phase comprend la constitution du groupe, la définition du concept du jeu, la rédaction et la validation du cahier des spécifications techniques, ainsi que les premières recherches techniques et expérimentations. Une première preuve de concept est présentée lors de la première soutenance, attestant de la faisabilité technique du projet.

#### Phase 2 – Développement principal (entre la première et la deuxième soutenance)

Cette phase est dédiée à l'implémentation des fonctionnalités principales du jeu. Elle inclut notamment le développement du multijoueur en réseau, de l'intelligence artificielle, du gameplay central, du level design et des premiers éléments sonores. Lors de la deuxième soutenance, une version avancée et partiellement finalisée du jeu est présentée, montrant une progression significative du projet.

#### Phase 3 – Finalisation et stabilisation (jusqu'à la soutenance finale)

La dernière phase vise à finaliser l'ensemble des fonctionnalités prévues, corriger les bugs, améliorer la stabilité et l'ergonomie du jeu, intégrer les contenus définitifs et préparer les

livrables finaux. La soutenance finale marque la validation du projet à travers la présentation d'un jeu complet, fonctionnel et conforme aux spécifications validées.

## **Estimation des ressources:**

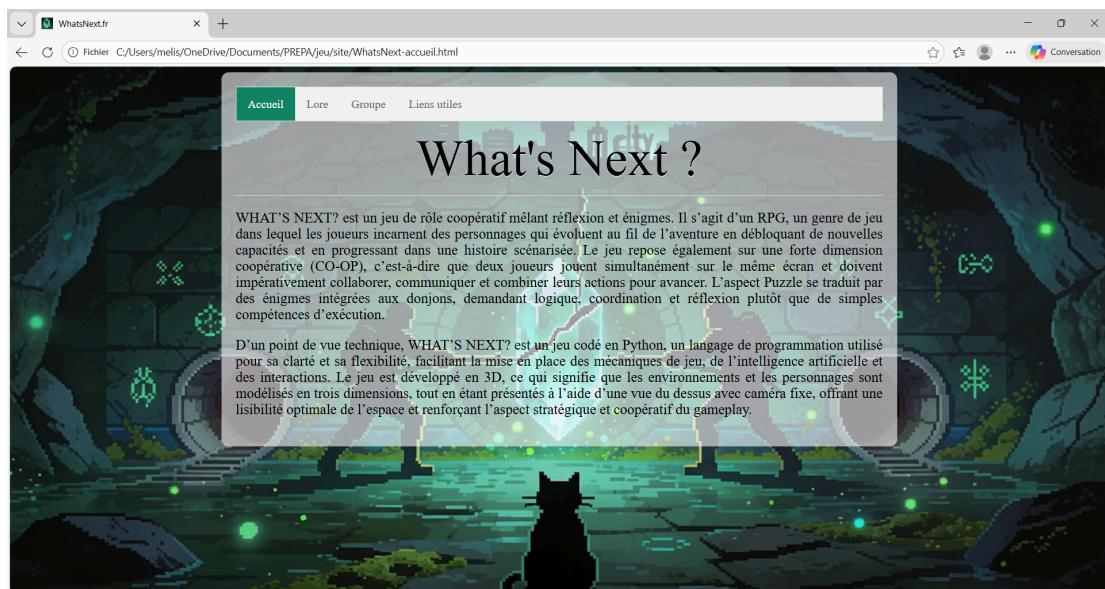
Le projet repose principalement sur des ressources humaines, mobilisant l'ensemble des membres du groupe sur toute la durée du projet. Les tâches sont réparties entre les membres selon un système de responsables et de suppléants afin d'assurer la continuité du développement.

Aucune estimation budgétaire financière n'est prévue, le projet s'inscrivant dans un cadre pédagogique. Les ressources matérielles utilisées correspondent aux équipements personnels des membres du groupe (ordinateurs, outils logiciels gratuits ou open source).

## ANNEXE:

Ces visuels sont des concepts préliminaires susceptibles d'évoluer au cours du développement.

### Aperçu du site web:



### Aperçu des personnages:

