

Cahier des charges – Projet MAUJA Adventures

I Présentation du projet

1.1 Les acteurs du projet

Maître d'ouvrage : M. Provot Laurent

Maître d'œuvre :

Jérémy TREMBLAY,
Maxime WISSOCQ,
Ugo VIGNON,
Antoine VITON,
Adrien COUDOUR

Chef de projet : Jérémy TREMBLAY

Equipe :

Ugo VIGNON G3 → Créatif, maîtrise les bases de la programmation orientée objet.

Maxime WISSOCQ G4 → Entreprenant, rigoureux, maîtrise les bases de la programmation orientée objet.

Jérémy TREMBLAY G7 → Réalisation de plusieurs jeux-vidéos auparavant, planificateur

Adrien COUDOUR G7 → Documentation, maîtrise les bases de la programmation orientée objet.

Antoine VITON G7 → Rigoureux, organisé, a déjà réalisé un projet en programmation orientée objet.

1.2 Nature de la prestation demandée

1.2.1 Genèse de l'idée de projet

L'idée de ce projet est venue assez simplement. En effet, nous voulions développer un petit jeu-vidéo d'aventure et de réflexion car nous voulions explorer d'autres sentiers battus, découvrir de **nouveaux modes de conception**, et proposer un projet avec une utilisation concrète et **beaucoup de possibilités**. Avec un tel jeu, il nous est possible de proposer un moyen de **divertissement** aux personnes, même en temps de crise sanitaire. C'est Maxime WISSOCQ qui a eu l'idée de ce projet.

1.2.2 Description succincte du projet

L'idée est de réaliser un petit jeu-vidéo 2D d'**aventure** et de **réflexion**. Le personnage pourrait se déplacer, découvrir les différentes cartes composant le monde et donc explorer des environnements luxuriants, soignés et variés.

Le concept principal de ce projet serait de **résoudre des énigmes** assez poussées afin de progresser dans l'aventure. Pour étayer cet objectif, le joueur aura à sa disposition différents objets (bombes, arc...) afin de réaliser des actions pouvant modifier son environnement. Il pourra interagir dans de nombreuses circonstances (casser des jarres, pousser des caisses...).

Bien évidemment, il y aurait une part de **combat** dans ce jeu, avec différents ennemis, différentes attaques et objets, et un système de points de vie. L'ensemble des actions se dérouleraient dans une même fenêtre, et il y aurait une caméra vue du dessus (aérienne).

On pourra jouer à ce jeu à plusieurs joueurs, depuis plusieurs ordinateurs différents (2 joueurs minimum, jusqu'à 4). Il contiendra donc un système de **jeu en réseau local** pour pouvoir permettre de partager des parties avec d'autres utilisateurs.

Notre projet s'inspire de nombreux jeux déjà existants sur le marché, mais il est unique et ce pour plusieurs raisons. Tout d'abord, nous allons créer **nos propres environnements** et une ambiance visuelle qui nous ressemble. De plus, l'objectif est de réaliser un jeu en réseau sur un **système d'énigmes assez poussé** pour créer un *gameplay* innovant. Enfin, nous incorporons des éléments de jeu qui nous sont personnels (décrits plus bas).

Pour l'utilisateur, la finalité du produit est simplement de pouvoir **jouer au jeu** tout en divertissant, en réfléchissant et appréciant l'environnement. Pour le demandeur, la finalité est que le jeu soit **fonctionnel** d'une part, et aussi que toutes les fonctionnalités prévues soient présentes d'autre part (décrites plus loin).

Ce jeu s'intitulera *MAUJA Adventures*.



Figure 1.1: The Adventures of Alundra, 1997



Figure 1.2: A Link to the Past, 1991

Images tirées de quelques jeux qui nous inspirent, et dont on souhaiterait simuler un univers semblable.

1.2.3 Détails

Voici quelques informations supplémentaires relatives au jeu que nous souhaitons concevoir :

Genre : Jeu d'aventure et de réflexion multijoueur

Style : Aventure / Action / Enigme / RPG

Plate-forme : Ordinateur, Windows 10 / Linux

Nombre de Joueurs : de 1 à 4

Pegi : 7+

Moteur : Création du moteur de jeu par nous-même

Langage de programmation : Java, framework JavaFX

Outils : IDE IntelliJ, Tiled, versionneur de fichier Git

Inspirations :

- Binding of Isaac (pour l'ambiance)
- The Legend of Zelda (pour le gameplay)
- Stardew Valley (pour les graphismes)
- The Adventures of Alundra (pour les environnements)



Figure 1^{er}: Logo de notre jeu

1.2.4 Une fois le projet terminé

Quand bien même le projet terminé, nous pourrions toujours **ajouter plus d'éléments** dedans, créer de nouvelles cartes et une histoire, ajouter d'autres énigmes et interactions.

Il sera sûrement proposé en **open source** si des gens souhaitent un jour s'inspirer de notre travail. On peut également imaginer de le rendre **compatible sur les téléphones et tablettes** et de le déposer sur le *Play Store* pour qu'il devienne accessible à un maximum de personnes.

1.3 Estimation des grandes étapes et dates butoirs

1.3.1 Livrables attendus

En fin de période 2, donc à la date du 23/01/2022 nous devons avoir ces livrables de terminés et disponibles :

1. Le **personnage doit pouvoir se déplacer** à l'écran.
2. Le personnage doit pouvoir se déplacer sur plusieurs **cartes différentes**, toutes les cartes qui constituent le jeu final.
3. Le personnage doit être **soumis aux collisions**, et donc ne pas pouvoir traverser les obstacles.
4. La **caméra** doit suivre le personnage, être centrée sur lui.
5. Le joueur doit pouvoir **attaquer** et se **protéger**.
6. Des **ennemis** doivent être créés et disposés sur la carte. Il doit également y avoir des **Boss**. Le joueur doit pouvoir les attaquer, subir des dégâts, perdre.
7. Le joueur doit avoir à sa disposition les **différents objets** lui permettant d'interagir, d'attaquer et de résoudre des énigmes, tel que l'arc, les bombes, etc.
8. Le jeu doit s'adapter à la **taille de la fenêtre**.
9. Il doit y avoir un **menu principal** permettant de choisir de jouer en solitaire ou en multijoueur (pas d'implémentation du mode multijoueur encore).

En fin de période 3, donc à la date du 21/03/2022, nous devons avoir ces livrables de terminés et disponibles :

1. **Jeu en réseau local** à 2 joueurs **minimum** fonctionnel. Possibilité de jouer à plusieurs.
2. **Le personnage doit être animé** afin d'être plus beau visuellement, avoir des animations d'actions, etc.
3. Le personnage doit pouvoir interagir avec des leviers et boutons qui réaliserons certaines actions.
4. **Le personnage doit pouvoir interagir** avec l'environnement (jarres, caisses, tonneaux...)
5. Il doit y avoir une **interface au jeu** sur laquelle on voit les points de vie du personnage, ses objets et leur nombre, les clés qu'il possède...
6. Des **trésors et bonus** doivent être disséminés, et lâchés par les monstres (vie, éventuellement argent).
7. Le jeu en solitaire doit pouvoir être **sauvegardé**, la progression du joueur, ses objets, son avancée seront gardés en mémoire.
8. Des **pièges** seront ajoutés (piques, boules qui roulent) et qui feront perdre de la vie au personnage
9. Un **écran de Game Over** sera ajouté si le personnage perd, ou si tous les joueurs perdent en mode multijoueur.

D'autres livrables optionnels ont été définis au tout début du projet. Il a été défini qu'ils seront réalisés si on a le temps, et seront considérés comme des « bonus » s'ils sont réalisés :

1. **Animations des tuiles**, afin de donner une impression de mouvement, de rendre le jeu plus visuel
2. **Interactions liées aux projectiles** : afin de pouvoir renvoyer les projectiles des ennemis
3. **Menu de pause** : pour mettre le jeu en pause
4. **Chat textuel pour le jeu en réseau** : permet aux joueurs de discuter entre eux
5. **Personnages non-joueurs** : on peut leur parler, ils se déplacent arbitrairement
6. **Menu et système d'équipement** : permet de s'équiper de diverses tenues
7. **Menu d'inventaire** : permet de voir les objets accumulés jusqu'à présent
8. **Écran de création de personnage** : Permet de créer son personnage avant de commencer une partie
9. **Effets sonores et musiques** : pour rajouter une ambiance au jeu

1.3.2 Délai de réalisation

Le délai de réalisation total est de 18 semaines soit entre 4 et 5 mois. Et nous jalonnons notre projet grâce à des sprints d'une durée de 2 semaines chacun à la fin des quels nous devons produire une démo.

1.3.3 Niveau de confidentialité

Dans le cadre de ce projet, et concernant le fait que nous souhaitons sûrement laisser notre projet libre d'accès afin que n'importe qui puisse le reprendre et le modifier, nous n'imposons pas de niveau de confidentialité à ce projet. Aussi, le niveau NC (Non classifié) sera appliqué, nous ne réaliserons pas de rétention de données et partagerons le code source de ce projet à quiconque sur la plateforme Github.

1.3.4 Estimation du budget

Devis						
Désignation	Quantité	Prix unitaire HT	Prix total HT	Taux de TVA	TVA	Prix TTC
Main d'œuvre	1 000 (heures)	14,60 €	14 600,00 €	20 %	2 920,00 €	17 520,00 €
Ordinateur	5	24,75 €	123,75 €	20 %	24,75 €	148,50 €
Location local	4,5 (mois)	173,33 €	780,00 €	20 %	156,00 €	936,00 €
Charges diverses	4,5 (mois)	43,00 €	193,50 €	20 %	38,70 €	232,20 €
Charges sociales*	1 000 (heures)	4,38 €	4 380,00 €	20 %	876,00 €	5 256,00 €
TileSet (Ressources graphiques)	2	50,00 €	100,00 €	20 %	20,00 €	120,00 €
Total			20 177,25 €	20 %	4 035,45	24 212,70 €

Le projet va durer 18 semaines (=4,5 mois) sur cette période nous avons 3 clients, nous avons donc utilisé les 3 clients comme clef de répartition :

- **Ordinateur** a coûté 600 € l'unité avec un taux d'amortissement de 0,33 sur 3 ans : $600 \times 0,33 = 198$; $198/12=16,5$; $16,5 \times 4,5=74,25$; $74,25/3=24,75$ pour cette période nos clients nous doivent 24,75 € (par ordinateur et par client).

- **Loyer** 520 € par mois soit $520 / 3 = 173,33$ € par client.

- **Charges diverses** 129 € par mois soit $129 / 3 = 43$ € par client.

*Charges sociales 30 % du salaire brut $14,60 \times 30/100 = 4,38$ € par heure.

1.4 Organisation de l'équipe et du travail

Nous organisons notre travail en utilisant la méthode agile SCRUM. Une méthode de travail qui consiste à réaliser plusieurs sprints de travail, et de fournir des livrables au client à chaque fin de sprint par l'intermédiaire d'une démo.

Nous avons pour cela commencé par réaliser un backlog de produit, c'est à dire définir toutes les *histoires* à réaliser avec leur importance et la durée prévue. Ces histoires sont pour la plupart des

fonctionnalités. La méthode agile SCRUM, prévoit la division du projet en sprints comme évoqué précédemment. Ces périodes de travail ont une durée définie, un but, une liste d'histoires à réaliser et une liste des membres de l'équipe. Les histoires sont sélectionnées à partir des points d'histoires, basés sur les estimations réalisées (on prend les histoires qu'il est possible de réaliser durant le sprint, pas trop mais assez, en se basant sur la quantité de travail que l'on produit chaque semaine).

Nous nous réunissons le lundi avec le maître d'ouvrage (M. Provot Laurent) pour la réunion de début de sprint, réunion où nous définissons quelles tâches devront être réalisées lors du sprint et qui devra s'en charger. Ces réunions durent en moyenne de 1 à 2h. Nous avons décidé qu'en général, un sprint dure 2 semaines. Nous nous réunissons donc une fois toutes les deux semaines.

Nous réalisons en fin de semaine (généralement le vendredi), une démo où nous présentons les livrables réalisés lors du sprint au maître d'ouvrage. Ces démos sont assez courtes, elles ne durent que 30 minutes environ.

Les outils de communication que nous utilisons sont : Discord, un logiciel pour pouvoir communiquer via des chats écrits et vocaux, la plate-forme Gitlab permettant de bien communiquer sur l'avancée du projet avec une liste de tâches et fonctionnalités.

L'outil utilisé pour partager le travail que nous utilisons est Git, un versionneur de fichiers collaboratif pour piloter des dépôts de code source et gérer leurs différentes versions.



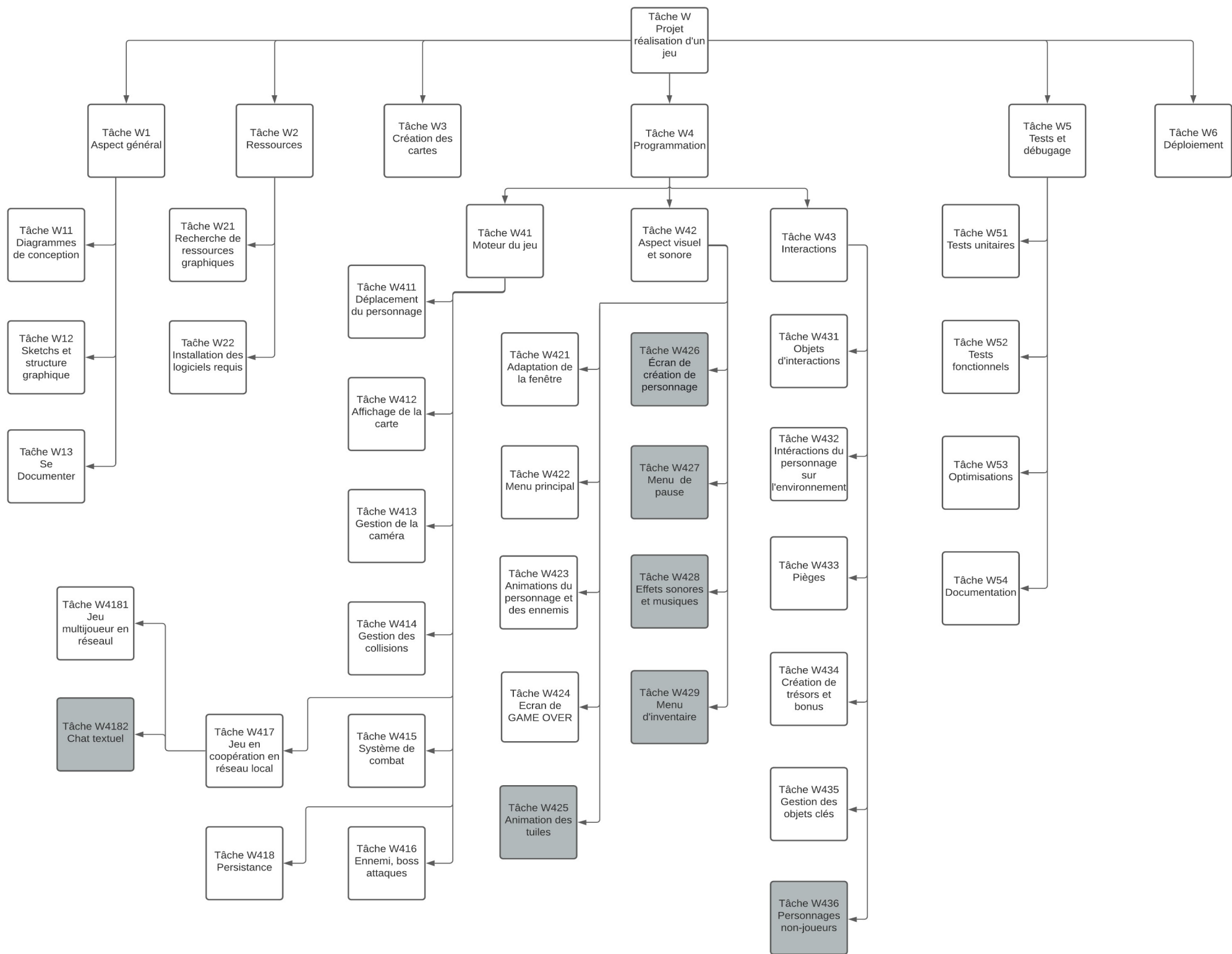
Résolution du bug sur la gestion des collisions	#13 · created 1 day ago by jetremblay	Bug	Important
Optimisation des classes Jeu, Fenetre et Decoupeur	#12 · created 1 day ago by jetremblay	Important	Optimisation
Limitation des mouvements du personnage	#10 · created 1 week ago by jetremblay	Tâche	🕒 2d
Gestion des collisions	#6 · created 1 week ago by jetremblay	Fonctionnalité	🕒 3d

Nous réalisons des mêlées régulières en vocal afin d'avancer, de planifier et de réaliser parfois du *live coding* : une personne programme en directe, en partage d'écran, et les autres l'aident directement.

Nous discutons beaucoup pour nous tenir au courant de l'avancée du projet et des problèmes.

II Cahier des charges fonctionnels

2.1 Fonctionnalités attendues



Légende :



: tâches optionnelles qui seront réalisées si nous avons le temps

2.2 Contraintes pesant sur la réalisation du projet

Les graphismes et images utilisées, et les éventuels sons et musiques qui seront utilisés dans le cadre de ce projet seront libres de droits, et respecterons donc la propriété intellectuelle des auteurs. Nous n'allons pas ajouter des images dont nous n'avons pas les droits et nous citerons les auteurs dans notre jeu si ceux-ci le réclament.

Le projet doit être terminé fin mars, plus précisément, le jeu doit être fonctionnel, complet, et disponible le 21 mars, et doit respecter les contraintes listées ci-dessous.

Pour un projet d'une telle envergure et d'un délai proche de 5 mois, le budget maximal donné qui ne devra pas être dépassé est de 25 000€ euros TTC.

2.3 Contraintes pesant sur l'utilisation du produit

Les contraintes suivantes ont été définies, et devront être respectées jusqu'à la livraison finale du projet :

- Le jeu doit être adaptable à la taille de l'écran, autrement dit, le jeu doit pouvoir fonctionner sur des écrans de taille différentes, et rester *responsive*, agréable d'utilisation.
- Le jeu doit être accessible, les contrôles doivent sembler logiques et simples d'utilisation.
- Le jeu doit être compatible sur les systèmes d'exploitation Linux et Windows 10
- Le jeu doit être déployé, pouvoir être joué en l'état.
- Le jeu doit être auto-suffisant, il ne doit nécessiter aucune installation d'un logiciel tiers.
- Le jeu doit être disponible à la date de fin prévue citée en 1.3, et fonctionner en l'état avec les fonctionnalités prioritaires.
- La partie qui concerne le jeu en réseau doit être sécurisée, il ne doit pas y avoir de faille de sécurité, il doit donc être penser avec le *Privacy By Default*.
- Aucune donnée personnelle, sensible ou confidentielle ne doit être stockée sur l'utilisateur final. Ainsi, même dans le cas du jeu en réseau, il ne faudra pas conserver de données relatives aux joueurs.

2.4 Critères d'appréciation de la qualité du produit

Le jeu livré doit être soigné, et doit posséder les spécificités énoncées :

- L'interface, les graphismes, le *gameplay* doivent avoir été pensés en amont, et doivent être correctement implémentés,
- Le jeu doit être beau visuellement sans pour autant être extravagant,
- Jeu en réseau local gérant les éventuels problèmes de connexion des joueurs : un problème de connexion d'un des joueurs ne doit pas impacter les autres joueurs,
- Les énigmes proposées doivent être originales, ni trop simples, ni trop difficiles,
- Les contrôles doivent être accessibles et ergonomiques,
- Le jeu doit être fluide sur tous les ordinateurs récents.