CAHIER DES CHARGES

Version : 1.2
Date : 10/10/2023

Client	Prestataire
	Nom 1 : Donovan Casagrande Nom 2 : Benoit Maurice Nom 3 : Benjamin Haurogne Nom 4 : Arthur Gros

1.1 Cahier des charges approuvé dans sa version
1.2 Le / / par

zone réservée

1 Introduction

1.1 Objet du document

Ce document décrit **tous** les services que doivent rendre le produit et ses livrables et toutes les exigences qu'ils doivent satisfaire.

1.2 Portée du document

Ce document est destiné à formaliser le besoin du client M. Kessler dans le cadre du projet Tamagotchi.

1.3 Terminologie

Terme	Description
Tamagotchi	Un Tamagotchi est un animal de compagnie virtuel japonais créé par la société japonaise Bandai en 1996. Tamagotchi est un mot-valise : « Tamago » pour œuf et « wotchi » pour montre (« watch » en anglais).
« Design Pattern »	Un « design pattern » ou patron de conception, est une manière de concevoir un code afin de le rendre plus modulable. C'est donc une pratique qui permet de résoudre un problème de conception d'un logiciel.
Produit	Terme générique désignant l'objet de la demande du client. Il recouvre aussi bien un système qu'un service, sans préjuger de la part de logiciel et de matériel intervenant dans la réalisation.
LibGDX	LibGDX est une bibliothèque utilisé pour le développement de jeux permettant aux développeurs de créer des jeux multi-plateformes.
Emulateur	Ensemble des outils permettant à un ordinateur de copier le fonctionnement d'une autre machine. Les émulateurs sont très utilisés pour jouer à de vieux jeux vidéo.
JAVA	JAVA est un langage de programmation de haut niveau orienté objet. Un logiciel écrit en langage JAVA a pour particularité d'être compilé vers un code intermédiaire formé de « Bytecodes » et qui peut être exécutée dans une machine virtuelle JAVA.

1.4 Abréviations

Abréviations	Signification	Libellé
MVC	Modèle-Vue-Contrôleur	Le MVC est un patron de conception (design pattern) utilisé lors du développement d'un logiciel. Il se décompose en 3 parties : M pour Modèle, V pour Vue et C pour Contrôleur. Il vise à organiser le code de manière à séparer les préoccupations liées à la gestion des données (géré par le Modèle), l'interface (géré par la Vue), et la logique de contrôle qui fait le lien entre le Modèle et la Vue (géré par le Contrôleur).

Abréviations	Signification	Libellé
UML	Unified Modeling Language	UML signifie « Unified Modeling Language ». Le langage UML est un moyen de représenter visuellement des systèmes logiciels et d'autres aspects de conception et d'analyse avec des notations graphiques et des conventions de modélisation. UML est un formalisme de modélisation.
JSON	JavaScript Object Notation	JSON signifie « JavaScript Object Notation ». JSON est un format standard utilisé pour représenter et sérialiser des données structurées et les échanger sur un réseau, généralement entre un serveur et des applications Web.
VIT	*Vitale*	*Exigences fonctionnelles ou non fonctionnelles indispensables*
IMP	*Importante*	*Exigences souhaitées mais non exigées*
MIN	*Mineure*	*Exigences non exigées immédiatement, mais qui devront être prises en compte ultérieurement par le produit (impact sur l'évolutivité)*

2 Les objectifs du produit

2.1 Définition du produit

Tamagotchi est un animal de compagnie virtuel japonais créé en 1996 par une société japonaise, Bandai. C'est le mot-valise d'œuf (« Tamago ») et montre (« Wotchi »).

C'est un jouet électronique portable ressemblant à un œuf. A l'intérieur de cet œuf virtuel se trouve un animal de compagnie virtuel dont il faut s'en occuper (nourrir, soigner, manger, dormir, etc…) afin de le maintenir en vie.

Le livrable peut être à la fois une application et un objet.
Les objectifs sont de répondre à une demande croissante d'une compagnie virtuel.
La demande est revenue sur le devant de la scène par effet de nostalgie des parents.

2.2 Contexte économique du produit

Le marché visé est principalement le marché des enfants et des collectionneurs. Les enseignes de jouets seront les principales vendeurs du produit.

La clientèle ciblé sont les enfants, qui ont entre 6 et 12 ans. Les enfants sont à la fois des garçons et des filles.

L'Asie, l'Amérique du Nord mais aussi l'Europe sont des zones très demandeuses de ce produit.

Bien que nous parlions principalement des enfants, qui sont la clientèle cible prioritaire, les adultes peuvent aussi être un marché intéressant grâce à l'effet nostalgique que produit l'objet en eux.

2.3 Contexte d'exploitation du produit

Bandai Namco est le principal créateur de Tamagotchi.
Il ne faut pour autant pas oublier que la contrefaçon existe et est une grande concurrence (illégale). En 1997, la concurrence illégale des contrefacteur leur a rapporté près de 13 millions d'euros.

Aujourd'hui en France, de nombreux sites en ligne vendent ce produit : Amazon, Carrefour, Cora, FNAC, JouéClub, La Grande Récré, PicWicToys.

Ces dernières années les émulateurs de Tamagotchi se sont multipliés, dont une majorité d'entre eux sont gratuits.
Enfin, les dernières concurrences possibles avec nous sont les autres groupes.

3 Exigences sur le produit

3.1 Capacités fonctionnelles

3.1.1 Description des fonctionnalités

Lorsque l'on allume le jeu, on a une page d'accueil qui s'affiche. (VIT)

Sur cette page d'accueil, il y a des boutons permettant :
- *De se déplacer dans une nouvelle page pour la création d'une nouvelle partie, (VIT)*
- *De quitter le jeu, (IMP)*
- *D'aller dans les paramètres. (MIN)*

La création du nouvelle partie emmène l'utilisateur dans une page où il devra personnaliser sont Tamagotchi. (VIT)

Il y aura la sélection du personnage parmi le dinosaure, le chat, le chien et le robot à faire en cliquant dessus. (VIT)
Il y aura une zone de texte en bas de l'écran pour mettre un nom au Tamagotchi. (MIN)

Le chat, le chien et le dinosaure ont des boutons pour chaque actions :
- *Eat, (VIT)*
- *Sleep, (VIT)*
- *Wash, (VIT)*
- *Play, (IMP)*
- *Work, (MIN)*
- *Pay, (MIN)*
- *Scream : Meow ou Bark ou Roar. (MIN)*

Le robot aura des boutons avec des actions uniques :
- *Maintain, équivalent de Wash (VIT)*
- *Reload, équivalent de Sleep (VIT)*
- *Repair, (VIT)*
- *Play, (IMP)*
- *Work, (MIN)*

- *Pay, (MIN)*
- *Dance. (MIN)*

Il y aura différentes pièces dans lequel l'utilisateur pourra se déplacer (salle de bain, cuisine, salon, et jardin).

3.1.2 Interopérabilité

L'interface et les fonctionnalités sont améliorées par l'utilisation de la bibliothèque LibGDX. L'application sera codée en JAVA.
La sauvegarde des parties sera gérée par des fichiers JSON.

L'application sera optimisée de sorte qu'un maximum d'ordinateur puisse le supporter, qu'ils soient portables ou fixes, qu'ils soient bas de gamme ou haut de gamme. Les systèmes d'exploitation visés sont Linux et Windows.

3.1.3 Conformité réglementaire

L'âge minimum conseillé est de 7 ans. Il suit la réglementation PEGI 7 signifiant : " Le jeu est destiné aux enfants de plus de 7 ans. Il comporte des scènes effrayantes pour les très jeunes enfants".
Le produit doit se conformer aux articles de la loi Informatique et liberté.

Si l'application venait à être créer en objet physique, il devrait suivre ces réglementations :

Le constructeur déclare que l'équipement de type SWALLOWUNI est conforme à la directive 2014/53/UE.
Le constructeur déclare que l'équipement est conforme au texte intégral de la déclaration de conformité de l'UE.
L'appareil est testé et jugé conforme à la partie 15 des règles et les limites de classe B de la FCC (« Federal Communications Commission ») .
Il y a aussi une politique de confidentialité qui s'applique.

En tant que concepteur, nous prendrons en compte que l'utilisateur puisse mettre son nom et/ou son prénom, le nom et/ou prénom de l'un de ses proches, ou d'autres informations personnelles pour le pseudo de son Tamagotchi. Nous proposerons automatiquement des pseudos originaux afin de diriger l'utilisateur vers le choix d'un pseudo impersonnel.

3.2 Exigences non fonctionnelles

3.2.1 Fiabilité

Lorsque l'on clique sur le personnage de façon frénétique, il doit réagir le maximum de fois qu'il peut.

Si la sauvegarde manuel ne marche pas, il devrait y avoir une sauvegarde automatique.

Si le joueur veut fermer le jeu rapidement sans sauvegarder, il devrait y avoir un message.

Afin d'éviter le « spam click » (sur-cliquage), il y aura un délai de quelques secondes avant qu'il puisse faire ou refaire une action, comme travailler par exemple.

Si l'ordinateur de l'utilisateur devient inutilisable et qu'il perd notre jeu, nous nous engageons à renvoyer l'application identique à celui que l'utilisateur avait.

Il y aura des méthodes de tests et de vérifications afin de contrôler le bon fonctionnement des attributs.

3.2.2 Sécurité

La sauvegarde des parties ne posera pas de problème particulier car c'est un jeu qui ne se joue pas en ligne. La mise en place de fichier JSON est la solution choisi pour la sauvegarde des parties et donc pour sauvegarder les données de l'utilisateur.

Les fichiers pourront être en format de compression « .jar » et les rendre impossible à décompiler.
Face à un utilisateur accompli en informatique, il y a une basse probabilité qu'il puisse récupérer le code source et le déchiffrer.

Il peut y avoir aucun intermédiaire entre l'entreprise, soit nous, et l'utilisateur. Nous pouvons donc envoyer le produit directement à l'utilisateur.

3.2.3 Facilité d'utilisation

Il y aura :

- *Un onglet pour les règles du jeu qui sera accessible dès la page d'accueil,*
- *Un tutoriel si besoin pour le joueur débutant en début de partie,*
- *Le choix des niveaux qui sera directement proposé lors de la création de la partie,*
- *Des images explicites (par exemple une image d'un cœur à côté de la barre de vie).*

Un joueur qui n'a jamais connu de jeux vidéo ou qui n'est pas un utilisateur régulier de Tamagotchi, peut prendre le jeu en main rapidement.

3.2.4 Rendement

L'utilisateur est unique dans le cadre du Tamagotchi. Mais le nombre d'utilisateur différent sur cette même application pourra être au maximum de 10, sinon nous prenons en compte que 1 utilisateur équivaut à une partie car il y aura jusqu'à 10 parties sauvegardées au maximum.

L'application prend à ce jour, 20Mo d'espace sur le disque.
Le jeu tourne à 30 images par seconde, c'est-à-dire qu'il y a 30 réactualisations de l'image par seconde. Plus le nombre d'images par seconde est élevé, plus le mouvement apparaît fluide devant les yeux.

Les calculs concernant la gestion des attributs du Tamagotchi sont assez simples, afin que tout ordinateur puisse supporter le logiciel.

3.2.5 Maintenabilité

Nous utiliserons le modèle MVC, qui est un design pattern ayant la particularité de bien séparer les données de l'application, l'interface et la classe qui gère le lien entre les 2. Cela facilitera la maintenance, les erreurs et leur gestion seront prises en charge plus facilement.
Grâce au modèle MVC, l'évolution du logiciel pourra se faire plus simplement, et si un nouveau développeur est intégré au projet , il pourra apprendre et comprendre le code et les fonctionnalités du logiciel plus vite.

3.2.6 Portabilité

Il y aura un document PDF joint au projet afin que tout utilisateur puisse ouvrir le projet sur n'importe quel système d'exploitation.

3.3 Exigences concernant le développement du produit

3.3.1 Objectifs de délais

Le cahier des charges v1.0 sera rendu dès le mois d'Octobre. Le 19 Octobre.

Du 07/09 au 14/09 : Analyse des besoins et définition des recettes.
Du 14/09 au 28/09 : Conception de l'architecture du logiciel.
Du 28/09 au 05/10 : Définition des tests d'intégration.
Du 12/10 au 19/10 : Conception des données de l'application.

3.3.2 Objectifs de coûts

2 chefs de projet : Gros Arthur et Haurogne Benjamin
Salaire des 2 chefs de projet : 4300 euros par mois

2 analystes de besoins : Maurice Benoit et Casagrande Donovan
Salaire des 2 analystes de besoins (analyste informatique) : 3125 euros par mois

Architecte logiciel : Casagrande Donovan
Salaire de l'architecte logiciel : 4500 euros par mois

2 développeurs seniors : Maurice Benoit et Gros Arthur
Salaire des 2 développeurs seniors : 4167 euros par mois

Gestionnaire de la qualité/validation : Haurogne Benjamin
Salaire du gestionnaire de la qualité/validation : 2333 euros par mois

Cependant, dans notre équipe de collaboration, tous les membres ont une double casquette, donc :

Salaire de Casagrande Donovan : 3125 + 400 euros pour sa compétence/double casquette d'Architecte Logiciel par mois (3525 euros)

Salaire de Haurogne Benjamin : 4300 + 300 euros pour sa double casquette de gestionnaire de la qualité/validation (4600 euros)

Salaire de Maurice Benoit : 3125 + 350 euros pour sa compétence/double casquette Développeur Senior par mois (3475 euros)

Salaire de Gros Arthur : 4300 + 350 euros pour sa compétence/double casquette Développeur Senior par mois (4650 euros)

Travaillant en distanciel, notre équipe n'a pas de charges administratives (loyer du bâtiment, etc.). Le matériel physique est à la charge des salariés.
Nos logiciels utilisés pour coder sont gratuits, il en va de même pour toute les bibliothèques utilisées.

En prenant en compte toutes ces contraintes, nous sommes à 48 750 euros sur les 3 mois du projets en charge salariale.
La masse salariale peut dépendre du nombre de collaborateur présent tout au long du projet.

3.3.3 Exigences de réalisation

Il faut que :

- *Ce soit coder de préférence en JAVA, sous un modèle MVC,*
- *Il y ait une interface graphique.*

Il doit impérativement y avoir un robot.
La fonction vitale du Tamagotchi est la prise en compte de la mort de celui-ci s'il n'est pas entretenu.

4 Synthèse des Exigences

4.1 Hiérarchisation des exigences fonctionnelles

Nom fonction	**Importance**
Mort (« Game Over »)	VIT
Actions possibles => Eat(), Sleep(), Wash(), Happiness()	VIT
Actions possibles => Repair(), Maintain(), Reload()	VIT
Créer nouvelle partie	VIT
Sauvegarder	VIT
Quitter le jeu	VIT
Changer de pièce	VIT
Choix du Tamagotchi	IMP
Choix de la difficulté	MIN

4.2 Hiérarchisation des exigences non fonctionnelles

Exigence	**Importance**
Actions possibles => Work(), Pay()	IMP

Design de l’application	MIN
Actions possibles => BuyApple(), BuyGoldenApple()	IMP
Actions possibles => BuyElexir(), BuySuperElexir()	IMP
Cri => Meow(), Bark(), Roar()	MIN

Choix du nom	MIN
Sons	MIN