

CAHIER DES CHARGES

Projet Tamagotchi - L3 MIS INFO UBS

Version :1.1

Date : 29/11/2021

<i>Client</i>	<i>Prestataire</i>
M. KESSLER	Nom1 : DIBERDER Evan nom2 : COSNIER Quentin nom3 : BUAN Kilian nom4 : REGNIER Alix
Cahier des charges approuvé dans sa version 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
Fonctionnalité	Ensemble des caractères ou des propriétés qui font que quelque chose remplit bien sa fonction.
Manger	Action réalisable par le tamagotchi. Manger permet au Tamagotchi de réduire sa faim.
Faire ses besoins	Action réalisable par le tamagotchi. Faire ses besoins permet au Tamagotchi d'augmenter son hygiène.
Jouer	Action réalisable par le tamagotchi. Jouer permet au Tamagotchi d'augmenter son bonheur.
Se laver	Action réalisable par le tamagotchi. Se laver permet au Tamagotchi d'augmenter son bonheur et son hygiène.
Dormir	Action réalisable par le tamagotchi. Dormir permet au Tamagotchi de réduire sa fatigue.

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.
Livrable	Ensemble des documents devant être livrés à une date donnée.
Maîtrise d'ouvrage	La maîtrise d'ouvrage, aussi dénommée maître d'ouvrage, est la personne pour qui est réalisé le projet.
Maîtrise d'œuvre	La maîtrise d'œuvre ou maître d'œuvre (souvent abrégée MOE ou Mœ ou Moe ou moe) est la personne physique ou morale choisie par le maître d'ouvrage pour la conduite opérationnelle des travaux en matière de coûts, de délais et de choix techniques, le tout conformément à un contrat et un cahier des charges.
Bête à cornes	Diagramme d'analyse fonctionnelle du besoin.
RGAA	Référentiel général d'amélioration de l'accessibilité. Le règlement national que doit (en théorie) respecter chaque logiciel en France. https://www.numerique.gouv.fr/uploads/rgaa/RGAA-v4.1.pdf

1.4 Abréviations

Abréviations	Signification	Libellé
FP	Fonctionnalité principale	Regroupe l'ensemble des fonctionnalités requises pour satisfaire la demande du client.
F-C	Fonctionnalité de conformité réglementaire	Regroupe l'ensemble des mesures et engagements pris afin que le logiciel soit conforme à la loi.
F	Fonctionnalité de fiabilité	Regroupe l'ensemble des mesures et engagements pris pour s'assurer du bon fonctionnement du logiciel.
FU	Fonctionnalité de facilité d'utilisation	Regroupe l'ensemble des mesures et engagements pris afin de rendre le logiciel utilisable par le plus de personnes possible.
M	Fonctionnalité de maintenabilité	Regroupe l'ensemble des mesures et engagements pris afin de faciliter la maintenance future du logiciel.
P	Fonctionnalité de portabilité	Regroupe l'ensemble des mesures et engagements pris afin de pouvoir utiliser le logiciel sur le plus d'appareils possibles.

1.5 Niveau d'importance

Niveau	Signification
5	<i>Exigences fonctionnelles ou non-fonctionnelles indispensables</i>
4	<i>Importance forte : fonctionnalités fortement souhaitées</i>
3	<i>Importance moyenne fonctionnalités utiles mais non-indispensables</i>
2	<i>Fonctionnalités d'importance faible</i>
1	<i>Fonctionnalités d'importance très faible</i>

2 LES OBJECTIFS DU PRODUIT

2.1 Définition du produit

Un Tamagotchi est un animal de compagnie virtuel japonais, créé en 1996 par la société japonaise Bandai. Son nom est un mot-valise créé à partir des mots « œuf » et « montre ». Le jeu consiste à simuler l'éducation d'un animal à l'aide d'une console miniature, de la taille d'une montre, dotée d'un programme informatique.

Ici, il faut adapter ce jeu au format logiciel de bureau. Les interactions se feront par souris et clavier.

2.2 Contexte économique du produit

M. Kessler demande dans le cadre du cours Génie Logiciel, un jeu de Tamagotchi développé par ses étudiants en troisième année de licence informatique. Il n'y a pas d'enjeu économique, le client est le fils de M. Kessler. Il pourra aussi servir à d'autres enfants, notamment les petits frères et sœurs des étudiants.

À titre de comparaison, il est aujourd'hui possible d'acheter des jeux de type Tamagotchi sur Amazon. Leurs prix peuvent varier entre 5 et plus de 100 euros en fonction des fonctionnalités proposées.

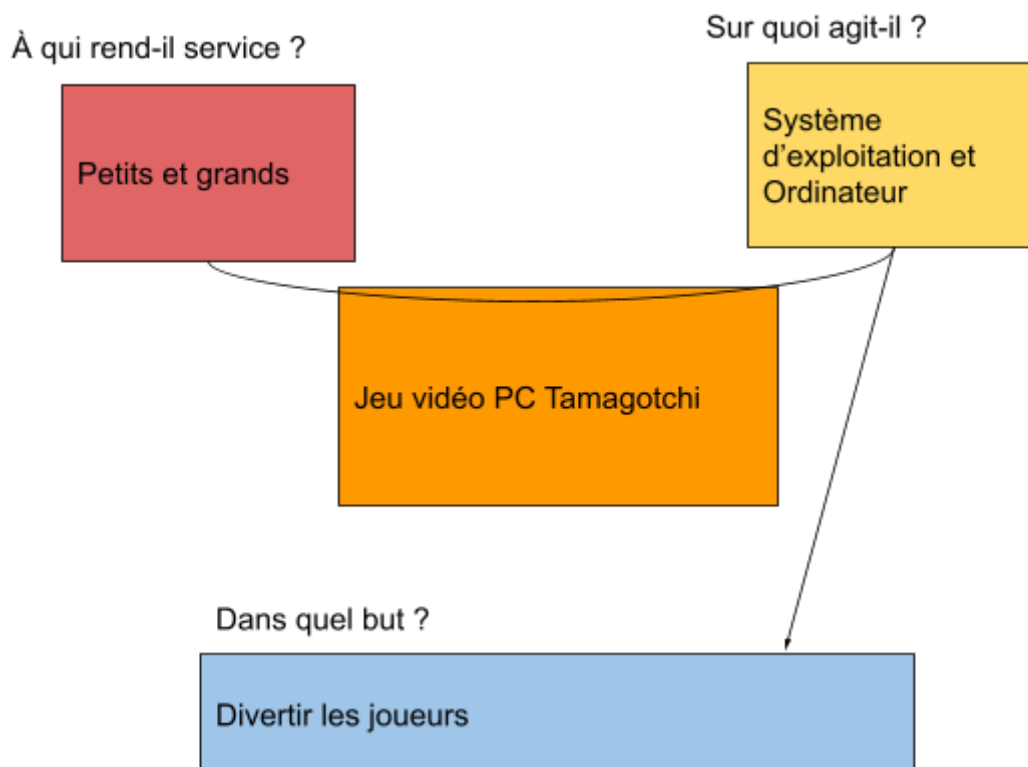
Les jeux Tamagotchi vendus sur le marché sont des jeux de type "pixel arté souvent en noir et blanc. Donc assez différent du jeu sur PC que nous allons développer.

2.3 Contexte d'exploitation du produit

Le produit sera principalement exploité par des enfants, et plus particulièrement par le fils de 12 ans de M. Kessler.

Ce public cible doit comprendre et appréhender le produit, le produit doit donc être adapté à ce public. bien vu ça

2.4 Bête à cornes résumant l'expression du besoin client



3 EXIGENCES SUR LE PRODUIT

3.1 Capacités Fonctionnelles

3.1.1 Description des fonctionnalités

Un tamagotchi avec des caractéristiques évoluant au cours du temps;

Un choix entre plusieurs types de créatures Tamagotchi doit être implémenté. Les créatures sont :

- *Un lapin*
- *Un chat*
- *Un chien*
- et, très important :*
- *Un robot*

On peut satisfaire ses besoins à l'aide de différentes actions faisant évoluer son état de santé.

Le Tamagotchi doit pouvoir se déplacer dans différentes pièces de sa maison et ne peut effectuer certaines actions que dans les pièces en rapport avec l'action.

Le Tamagotchi ne peut se déplacer qu'entre deux pièces directement connectées.

Le logiciel doit pouvoir sauvegarder jusqu'à trois parties que l'on doit pouvoir reprendre à n'importe quel moment, le Tamagotchi doit évoluer même pendant notre absence. Une base de données va permettre de stocker les sauvegardes.

3.1.2 Conformité réglementaire

Respect du PEGI 7:

Interdiction:

- *à la violence;*
- *au crime;*
- *à l'incitation à l'usage, à la détention ou au trafic de stupéfiants;*
- *à l'incitation à la consommation excessive d'alcool;*
- *ou à la discrimination ou à la haine contre une personne déterminée ou un groupe de personnes;*
- *Son support doit faire l'objet d'une signalétique destinée à en limiter la mise à disposition à certaines catégories de mineurs, en fonction de leur âge, et préciser le risque contenu dans le jeu vidéo.*

3.2 Exigences non fonctionnelles

3.2.1 Fiabilité

Le logiciel doit posséder un jeu de tests complet pour s'assurer de la fiabilité du logiciel. Il doit être joué automatiquement à chaque modification du logiciel (chaque "push" sur le dépôt du logiciel)

3.2.2 Facilité d'utilisation

Le logiciel doit être adapté à tous les publics et donc l'interface utilisateur doit être facile d'utilisation.

Il doit notamment respecter la norme d'accessibilité française sur les sites et logiciels : le RGAA.

Le jeu doit posséder plusieurs niveaux de difficultés :

Niveau facile : Accès aux statistiques, une action satisfait beaucoup le tamagotchi.

Niveau intermédiaire : Pas accès aux statistiques (uniquement aux états), une action satisfait bien le tamagotchi

Niveau difficile : Pas accès aux statistiques (uniquement aux états), une action satisfait difficilement le tamagotchi.

3.2.3 Maintenabilité

Le logiciel doit être commenté et lisible pour pouvoir être maintenu par la suite par l'équipe initiale ou une nouvelle équipe de développement.

3.2.4 Portabilité

Le logiciel doit pouvoir être lancé sur n'importe quel système d'exploitation de bureau, c'est-à-dire macOS, Windows et GNU+Linux.

3.3 Exigences concernant le développement du produit

3.3.1 Objectifs de délais

En tant qu'équipe de projet, un diagramme de Gantt avec les jalons est nécessaire.

Le projet doit être rendu en janvier.

Liste des jalons du projet :

- Cahier des charges, Spécifications et tests initiaux : 27 octobre 2021
- Premier livrable (conception) : 22 Novembre 2021
- Second livrable (conception + code + tests) : 4 janvier 2022

3.3.2 Exigences de réalisation

Le projet possède 5 acteurs du développement.

Tout d'abord, du côté de la maîtrise d'ouvrage, nous avons le client.

Celui-ci dialogue avec le chef de projet et l'analyste concepteur qui vont traduire la demande en conception de logiciel.

Ils font partie de la maîtrise d'œuvre. Le travail du chef de projet et de l'analyste concepteur va permettre au développeur senior et au gestionnaire des tests de programmer le logiciel.

Le développeur senior et le gestionnaire des tests participent à la conception et donnent leur expertise.

Lors de la phase de programmation, le chef de projet et l'analyste concepteur participent aussi au code, mais ils ne sont pas responsables.

Une exigence importante est la réalisation des tests lors de la phase de conception. Même s'ils pourront amener à évoluer par la suite.

4 SYNTHÈSE DES EXIGENCES

4.1 Hiérarchisation des exigences fonctionnelles

Type	Nom fonction	Importance
------	--------------	------------

FP	<i>Un tamagotchi avec des caractéristiques évoluant au cours du temps</i>	5
FP	<i>On peut satisfaire ses besoins à l'aide de différentes actions faisant évoluer son état de santé</i>	5
FP	<i>Le tamagotchi doit pouvoir se déplacer dans différentes pièces de sa maison et ne peut effectuer certaines actions que dans les pièces en rapport avec l'action.</i>	4
FP	<i>Le tamagotchi ne peut se déplacer qu'entre 2 pièces directement connectées.</i>	4
FP	<i>Le logiciel doit pouvoir sauvegarder jusqu'à 3 parties et on doit pouvoir reprendre à n'importe quel moment, le tamagotchi doit évoluer même pendant notre absence.</i>	3
FP	<i>Un choix entre plusieurs types de créatures Tamagotchi doit être implémenté. (Chien, chat, lapin et robot)</i>	4
C	<i>Respect du PEGI 7</i>	2

4.2 Hiérarchisation des exigences non fonctionnelles

Type	Nom exigence	Importance
F	Jeux de tests complets pour s'assurer de la fiabilité du logiciel	3
FU	<i>Le logiciel doit être adapté à tous les publics et donc l'interface utilisateur doit être facile d'utilisation</i>	3
FU	<i>Respect de la norme d'accessibilité : le RGAA.</i>	2
FU	<i>Plusieurs niveaux de difficultés</i>	2
M	<i>Le logiciel doit être commenté et lisible pour pouvoir être maintenu par la suite par l'équipe initiale ou une nouvelle équipe de développement.</i>	3
P	<i>Le logiciel doit pouvoir être lancé sur n'importe quel système d'exploitation de bureau, c'est-à-dire macOS, Windows et GNU+Linux.</i>	2