

# Cas d'utilisation et cas de tests 1 :

Élém. de cas d'utilisation	Description spécifique au projet
<b>Cas d'util.</b>	Création d'une partie
<b>Acteurs</b>	Le joueur
<i>Év. déclencheur</i>	Les 3 conditions sont remplies et la partie peut se lancer
<i>Parties prenantes et leurs intérêts</i>	Le joueur / Création du Tamagotchi
<i>Niveau</i>	Elevé
<i>Portée</i>	<ul style="list-style-type: none"> <li>• Choix du personnage parmi le chien, le chat, le dinosaure et le robot</li> <li>• Choix du pseudo du personnage</li> <li>• Choix du niveau de difficulté de la partie</li> </ul>
<i>Préconditions</i>	Que le joueur ait bien rempli les 3 conditions
<i>Post-conditions</i>	Les informations rentrées par le joueur soient enregistrées
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1. Le joueur ouvre le jeu</li> <li>2. Il clique sur le bouton « play »</li> <li>3. Il choisit son personnage, il lui donne un nom et il choisit le niveau de difficulté</li> <li>4. Il clique sur « jouer »</li> </ol>
<b>Extensions</b>	<ol style="list-style-type: none"> <li>1. Le joueur ne choisit pas de personnage</li> <li>2. Le joueur n'entre pas le nom du Tamagotchi</li> <li>3. Le joueur ne choisit pas le niveau de difficulté</li> </ol>
<i>Contraintes</i>	Il faut que le joueur prenne connaissance des règles du jeu avant de lancer une partie

Élém. de test de cas d'util.	Description spécifique au projet
<b>Cas de test</b>	Test du choix du pseudo
<b>Acteurs</b>	Utilisateur
<i>Év. déclencheur</i>	Champ de texte que l'on peut remplir
<i>Parties prenantes et leurs intérêts</i>	Utilisateur – Personnalisation du Tamagotchi
<i>Niveau</i>	Mineur
<i>Portée</i>	Le nombre de caractères minimum (1) et maximum (64) ainsi que la gestion des caractères spéciaux (pas de « & », « @ », « + », etc.
<i>Préconditions</i>	L'utilisateur doit créer une nouvelle partie et sélectionner son Tamagotchi
<i>Post-conditions</i>	L'utilisateur a un Tamagotchi au nom personnalisé
<b>Scénario de test</b>	<ol style="list-style-type: none"> <li>1. Toutes les données d'entrée vides (test limite/extrême) : <ol style="list-style-type: none"> <li>a. <u>Nom du Tamagotchi</u> = « » (chaîne vide)</li> <li>b. Réponse attendue : Un message d'erreur demandant à l'utilisateur de mettre au moins 1 caractère</li> </ol> </li> </ol>

	<p>2. Le nom donné au Tamagotchi est trop long :</p> <ul style="list-style-type: none"> <li>a. <u>Nom du Tamagotchi a plus de 64 caractères</u></li> <li>b. Réponse attendue : Un message d'erreur demandant à l'utilisateur de mettre un nom plus petit</li> </ul> <p>3. Le nom du Tamagotchi comporte des caractères spéciaux :</p> <ul style="list-style-type: none"> <li>a. <u>Nom du Tamagotchi comporte un « &amp; » ou un « @ » ou encore des symboles mathématiques</u> (par exemple)</li> <li>b. Réponse attendue : Un message d'erreur demandant à l'utilisateur de ne pas mettre de caractères spéciaux avec la précision dans le message d'erreur des caractères spéciaux à éviter</li> </ul>
<b>Extensions</b>	Cas de test du choix de l'animal
<b>Contraintes</b>	En lien avec d'autres cas d'utilisation du lancement d'une partie

Élém. de test de cas d'util.	Description spécifique au projet
<b>Cas de test</b>	Test du choix du personnage
<b>Acteurs</b>	Utilisateur
<b>Év. déclencheur</b>	Cliquer avec la souris sur l'un des personnages
<b>Parties prenantes et leurs intérêts</b>	Utilisateur – Personnalisation du Tamagotchi
<b>Niveau</b>	Mineur
<b>Portée</b>	Choix d'un personnage parmi 4 différents.
<b>Préconditions</b>	L'utilisateur doit créer une nouvelle partie
<b>Post-conditions</b>	L'utilisateur a un Tamagotchi
<b>Scénario de test</b>	<p>1. Aucun Tamagotchi sélectionné :</p> <ul style="list-style-type: none"> <li>a. Réponse attendue : Tamagotchi sélectionné par défaut par le jeu</li> </ul> <p>2. Un Tamagotchi est sélectionné :</p> <ul style="list-style-type: none"> <li>a. Réponse attendue : Le Tamagotchi est correctement sélectionné</li> </ul>
<b>Extensions</b>	Cas de test du choix du nom du Tamagotchi
<b>Contraintes</b>	En lien avec d'autres cas d'utilisation du lancement d'une partie

## Cas d'utilisation et cas de test 2 :

Élém. de cas d'utilisation	Description spécifique au projet
<b>Cas d'util.</b>	Reprendre une partie en cours
<b>Acteurs</b>	Le joueur
<i>Év. déclencheur</i>	Le joueur sélectionne la partie qu'il a enregistré précédemment
<i>Parties prenantes et leurs intérêts</i>	Le joueur et les parties sauvegardées
<i>Niveau</i>	Élevé
<i>Portée</i>	Choix de la partie enregistré
<i>Préconditions</i>	Le joueur doit au moins avoir créé une partie, pour pouvoir la reprendre
<i>Post-conditions</i>	La partie enregistré doit reprendre au moment où le joueur l'a enregistré
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1. Le joueur clique sur le bouton « jouer »</li> <li>2. Ensuite cliquer sur « parties sauvegardées »</li> <li>3. Cliquer sur la sauvegarde de la partie qu'on souhaite reprendre</li> <li>4. Reprendre la partie</li> </ol>
<b>Extensions</b>	<ol style="list-style-type: none"> <li>1. Aucune partie n'a été enregistré</li> <li>2. La partie enregistré a subi des détériorations de ces données ou a mal été enregistré</li> </ol>
<i>Contraintes</i>	Le joueur n'a pas encore créé de parties, ou alors la partie sauvegardée a subi des altérations sur ces données

Élém. de test de cas d'util.	Description spécifique au projet
<b>Cas de test</b>	Test d'une sélection d'une sauvegarde
<b>Acteurs</b>	Utilisateur
<i>Év. déclencheur</i>	Clique sur le bouton « Parties sauvegardées » de la page d'accueil
<i>Parties prenantes et leurs intérêts</i>	Utilisateur – Choix d'une partie existante
<i>Niveau</i>	Important
<i>Portée</i>	Nombre de parties sauvegardées peut aller jusqu'à 10. Permet de choisir la sauvegarde que l'utilisateur veut.
<i>Préconditions</i>	L'utilisateur doit avoir déjà créé une partie
<i>Post-conditions</i>	L'utilisateur va pouvoir continuer à jouer sur une ancienne partie
<b>Scénario de test</b>	<ol style="list-style-type: none"> <li>1. Aucune partie n'existe déjà : <ol style="list-style-type: none"> <li>a. Réponse attendue : Un message informant l'utilisateur qu'il ne peut accéder à aucune partie existante puisqu'il n'en existe aucune</li> </ol> </li> <li>2. Partie existante altérée par un bug : <ol style="list-style-type: none"> <li>a. Réponse attendue : Un message informant l'utilisateur qu'il ne peut pas accéder à la partie parce qu'elle n'est plus jouable</li> </ol> </li> </ol>

<b>Extensions</b>	Aucune interaction avec d'autres cas de test
<i>Contraintes</i>	En lien avec d'autres cas d'utilisation du chargement d'une partie

## Cas d'utilisation et cas de test 3 :

Élém. de cas d'utilisation	Description spécifique au projet
<b>Cas d'util.</b>	Le joueur souhaite acheter de la nourriture
<b>Acteurs</b>	Le joueur
<i>Év. déclencheur</i>	Le joueur a assez d'argent pour acheter le produit et qu'il sélectionne le produit qu'il veut
<i>Parties prenantes et leurs intérêts</i>	Le joueur + Boutique => Achat d'un produit
<i>Niveau</i>	Moyen
<i>Portée</i>	<ul style="list-style-type: none"> <li>• Choix du produit</li> <li>• Quantité du produit voulu</li> </ul>
<i>Préconditions</i>	Le joueur a suffisamment d'argent pour acheter le(s) produit(s)
<i>Post-conditions</i>	Le(s) produit(s) soit ajouter au panier du personnage et que l'argent soit débité
<b>Scénario nominal</b>	1. Le joueur clique sur l'icône boutique 2. Le joueur sélectionne le produit 3. Le joueur clique sur l'icône « valider l'achat »
<b>Extensions</b>	Le joueur n'a pas assez d'argent pour acheter le produit
<i>Contraintes</i>	Le joueur veut acheter un produit mais il n'a pas assez d'argent pour acheter le produit

Élém. de test de cas d'util.	Description spécifique au projet
<b>Cas de test</b>	Test de la méthode qui permet de travailler (l'aspect financier) avec la méthode d'achat de pommes ou d'huiles
<b>Acteurs</b>	Utilisateur
<i>Év. déclencheur</i>	On clique sur le bouton « travailler » pour gagner de l'argent, puis nous achetons dans la boutique les pommes pour les animaux et l'huile pour le robot
<i>Parties prenantes et leurs intérêts</i>	Utilisateur – Permet de maintenir en vie l'animal ou le robot car sans nourriture les animaux et/ou le robot meurt.
<i>Niveau</i>	Vitale
<i>Portée</i>	Nombre minimal et maximal de pommes ou d'huiles possible en stock. Prise en compte de la faim lorsqu'il n'y a plus aucune nourriture.
<i>Préconditions</i>	L'utilisateur est dans sa partie Le Tamagotchi ne doit pas être mort
<i>Post-conditions</i>	L'utilisateur a plus ou moins de nourritures et de pièces en stock
<b>Scénario de test</b>	1. Il y a 0 ou pas assez de pièces : <ol style="list-style-type: none"> <li><u>Le nombre de pièce est de 0 et l'utilisateur veut acheter de la nourriture</u></li> <li><u>L'utilisateur n'a pas assez de pièces pour payer de la nourriture</u></li> <li>Réponse attendue : Un message informant l'utilisateur qu'il n'a pas assez d'argent pour pouvoir payer de la nourriture</li> </ol>

	<p>2. Il y a assez de pièces pour pouvoir acheter quelque chose :</p> <p>a. Réponse attendue : Message de confirmation avant que l'achat soit fait</p> <p>3. Le maximum de pièces possible à avoir en stock est atteint :</p> <p>a. Réponse attendue : Un message indiquant que le maximum est atteint</p> <p>4. Il n'y a plus de nourriture en stock :</p> <p>a. Réponse attendue : Un message conseillant de travailler</p> <p>5. Le maximum de pommes ou d'huiles possible à avoir en stock est atteint :</p> <p>a. 1<sup>ère</sup> réponse attendue : Un message indiquant que le maximum est atteint</p> <p>b. 2<sup>ème</sup> réponse attendue : Un message conseillant de donner à manger au Tamagotchi</p>
<b>Extensions</b>	Intéraction possible avec le cas de test d'alimentation du Tamagotchi
<i>Contraintes</i>	En lien avec le cas d'utilisation « Le joueur souhaite acheter de la nourriture »

## Cas d'utilisation et cas de test 4 :

Élém. de cas d'utilisation	Description spécifique au projet
<b>Cas d'util.</b>	Le joueur souhaite nourrir son tamagotchi
<b>Acteurs</b>	Le joueur
<i>Év. déclencheur</i>	Le panier du tamagotchi ne soit pas vide et que le tamagotchi ait faim
<i>Parties prenantes et leurs intérêts</i>	Le joueur + Panier => Nourrir le tamagotchi
<i>Niveau</i>	Moyen
<i>Portée</i>	<ul style="list-style-type: none"> <li>• Choix de nourriture (rare ou commune)</li> <li>• La quantité à donner</li> </ul>
<i>Préconditions</i>	Le panier du tamagotchi ne doit pas être vide
<i>Post-conditions</i>	La quantité donnée et le type de nourriture doivent être enlevé du panier, et l'indicateur de faim du tamagotchi doit augmenter
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1. Le joueur doit se rendre dans la cuisine</li> <li>2. Il doit cliquer sur le panier</li> <li>3. Il choisit le type et le nombre de nourriture</li> <li>4. Il clique sur l'icône manger</li> <li>5. Il quitte le panier</li> </ol>
<b>Extensions</b>	Le panier est vide
<i>Contraintes</i>	Le panier du tamagotchi doit contenir au moins un objet de nourriture pour pouvoir nourrir le tamagotchi

Élém. de test de cas d'util.	Description spécifique au projet
<b>Cas de test</b>	Test d'alimentation du Tamagotchi
<b>Acteurs</b>	Utilisateur
<i>Év. déclencheur</i>	On clique sur le bouton « manger » pour que l'utilisateur puisse nourrir son Tamagotchi
<i>Parties prenantes et leurs intérêts</i>	Utilisateur – Permet de maintenir en vie l'animal ou le robot car sans nourriture les animaux et/ou le robot meurt.
<i>Niveau</i>	Vitale
<i>Portée</i>	Prise en compte de la faim lorsqu'il n'a pas mangé depuis un certain temps.
<i>Préconditions</i>	L'utilisateur est dans sa partie Le Tamagotchi ne doit pas être mort
<i>Post-conditions</i>	L'utilisateur a perdu un peu de nourriture qu'il avait en stock
<b>Scénario de test</b>	<ol style="list-style-type: none"> <li>1. Il n'y a plus de nourriture en stock :               <ol style="list-style-type: none"> <li>a. Réponse attendue : Un message conseillant de travailler</li> </ol> </li> <li>2. La barre de faim atteint un seuil critique :               <ol style="list-style-type: none"> <li>a. Réponse attendue : Un message apparait pour dire à l'utilisateur que le Tamagotchi risque de mourir</li> </ol> </li> </ol>

<b>Extensions</b>	Intéraction possible avec le cas de test d'achat de pommes/huiles
<i>Contraintes</i>	En lien avec le cas d'utilisation « Nourrir le Tamagotchi »



## Cas d'utilisation 5 :

Élém. de cas d'utilisation	Description spécifique au projet
<b>Cas d'util.</b>	Le joueur souhaite faire travailler son tamagotchi
<b>Acteurs</b>	Le joueur
<i>Év. déclencheur</i>	Pour que le tamagotchi puisse travailler, il ne faut pas qu'il soit fatigué ou malheureux et qu'il soit en forme
<i>Parties prenantes et leurs intérêts</i>	Le joueur + Tamagotchi => Gagner de l'argent
<i>Niveau</i>	Moyen
<i>Portée</i>	Argent gagné et par conséquent l'achat de nourriture
<i>Préconditions</i>	Le tamagotchi ne doit pas être fatigué ou malheureux et qu'il soit en forme
<i>Post-conditions</i>	L'argent gagné est ajouté au porte-monnaie du tamagotchi et l'énergie dépensée doit être enlevé sur les indicateurs de vies (hygiène, fatigue et joie)
<b>Scénario nominal</b>	<ol style="list-style-type: none"><li>1. Le joueur clique sur l'icône travaille</li><li>2. Pendant que le tamagotchi travaille, l'argent est ajouté et ces indicateurs de vies baissent</li><li>3. Le joueur clique sur l'icône quitter pour que le tamagotchi arrête de travailler</li></ol>
<b>Extensions</b>	Le tamagotchi n'a pas assez d'énergie pour travailler
<i>Contraintes</i>	Le tamagotchi est trop épuisé pour pouvoir travailler