

Cas d'utilisation : Effacer toutes les données

- **Acteur** : Joueur (utilisateur du programme)
 - **Événement déclencheur** : L'utilisateur choisit de supprimer toutes les données du jeu.
 - **Parties prenantes et leurs intérêts** :
 - Utilisateur : Souhaite supprimer les données pour redémarrer le jeu ou effacer des parties enregistrées.
 - **Niveau** : Sous-fonctionnalité
 - **Portée** : Suppression des données enregistrées.
 - **Pré-conditions** : L'utilisateur a des données sauvegardées.
 - **Post-conditions** : Toutes les données sont supprimées. L'état de la partie en cours est sauvegardé.
 - **Scénario nominal** :
 1. L'utilisateur sélectionne l'option pour effacer toutes les données.
 2. Le système sauvegarde l'état de la partie en cours.
 3. Le système supprime toutes les données enregistrées.
 - **Extensions** : N/A
 - **Contraintes** : N/A
-

Cas d'utilisation : Consulter une partie précédente

- **Acteur** : Joueur (utilisateur du programme)
 - **Événement déclencheur** : L'utilisateur choisit de consulter une partie sauvegardée.
 - **Parties prenantes et leurs intérêts** :
 - Utilisateur : Souhaite revoir une partie sauvegardée.
 - **Niveau** : Sous-fonctionnalité
 - **Portée** : Consultation des parties sauvegardées.
 - **Pré-conditions** : Il existe une ou plusieurs parties sauvegardées.
 - **Post-conditions** : L'utilisateur peut consulter l'état de la partie sauvegardée.
 - **Scénario nominal** :
 1. L'utilisateur sélectionne une partie sauvegardée à consulter.
 2. Le système affiche la partie.
 3. Le système sauvegarde l'état de la partie en cours.
 - **Extensions** : N/A
 - **Contraintes** : N/A
-

Cas d'utilisation : Quitter le programme

- **Acteur** : Joueur (utilisateur du programme)
 - **Événement déclencheur** : L'utilisateur choisit de quitter l'application.
 - **Parties prenantes et leurs intérêts** :
 - Utilisateur : Souhaite quitter l'application après avoir terminé la session.
 - **Niveau** : Objectif utilisateur
 - **Portée** : Fermeture du programme.
 - **Pré-conditions** : Une partie est en cours.
 - **Post-conditions** : Le système sauvegarde l'état de la partie avant de fermer.
 - **Scénario nominal** :
 1. L'utilisateur choisit de quitter le programme.
 2. Le système sauvegarde l'état de la partie en cours.
 3. Le programme se ferme.
 - **Extensions** : N/A
 - **Contraintes** : N/A
-

Cas d'utilisation : Reprendre partie en cours

- **Acteur** : Joueur (utilisateur du programme)
 - **Événement déclencheur** : L'utilisateur veut reprendre une partie mise en pause ou interrompue.
 - **Parties prenantes et leurs intérêts** :
 - Utilisateur : Reprendre une partie en cours pour continuer à jouer.
 - **Niveau** : Objectif utilisateur
 - **Portée** : Reprise de la partie en cours.
 - **Pré-conditions** : Une partie a été mise en pause ou interrompue.
 - **Post-conditions** : La partie est relancée avec les données précédemment sauvegardées.
 - **Scénario nominal** :
 1. L'utilisateur sélectionne l'option pour reprendre la partie en cours.
 2. Le système charge les données de la partie sauvegardée.
 3. La partie reprend au même point.
 - **Extensions** : N/A
 - **Contraintes** : N/A
-

Cas d'utilisation : Lancer une nouvelle partie

- **Acteur** : Joueur (utilisateur du programme)
- **Événement déclencheur** : L'utilisateur choisit de lancer une nouvelle partie.
- **Parties prenantes et leurs intérêts** :
 - Utilisateur : Souhaite commencer une nouvelle partie.
- **Niveau** : Objectif utilisateur

- **Portée** : Démarrage d'une nouvelle partie.
 - **Pré-conditions** : N/A
 - **Post-conditions** : Une nouvelle partie est créée et l'ancienne partie est supprimée si elle existait.
 - **Scénario nominal** :
 1. L'utilisateur sélectionne l'option pour lancer une nouvelle partie.
 2. Le système vérifie s'il existe une partie en cours.
 3. Si une partie en cours existe, elle est supprimée.
 4. Le système démarre une nouvelle partie.
 - **Extensions** :
 - Si une partie en cours est détectée, l'utilisateur est invité à confirmer la suppression de la partie actuelle.
 - **Contraintes** : N/A
-

Cas d'utilisation : Poser une carte

- **Acteur** : Joueur (utilisateur du programme)
 - **Événement déclencheur** : L'utilisateur choisit de jouer une carte durant son tour.
 - **Parties prenantes et leurs intérêts** :
 - Utilisateur : Souhaite jouer une carte pour avancer dans le jeu.
 - **Niveau** : Sous-fonctionnalité
 - **Portée** : Gestion du tour de jeu.
 - **Pré-conditions** : Le joueur a une carte en main et c'est son tour de jouer.
 - **Post-conditions** : La carte est jouée et les actions correspondantes sont effectuées.
 - **Scénario nominal** :
 1. Le joueur choisit une carte à poser.
 2. Le système applique l'effet de la carte.
 3. Les actions du tour sont mises à jour (par exemple : attaque, parade, avancer, poser une botte, etc.).
 4. Le coup est enregistré.
 - **Extensions** :
 - Si la carte n'est pas jouable, le joueur est informé.
 - Le joueur peut décider de jeter une carte si aucune action n'est possible.
 - **Contraintes** : Respect des règles du jeu.
-

Cas d'utilisation : Jeter carte dans une fosse

- **Acteur** : Joueur (utilisateur du programme)
- **Événement déclencheur** : Le joueur choisit de jeter une carte qu'il ne souhaite pas jouer.
- **Parties prenantes et leurs intérêts** :
 - Utilisateur : Veut se débarrasser d'une carte inutile.
- **Niveau** : Sous-fonctionnalité
- **Portée** : Gestion du tour de jeu.
- **Pré-conditions** : Le joueur a une carte à jeter.
- **Post-conditions** : La carte est jetée dans la fosse et une nouvelle carte est piochée.
- **Scénario nominal** :
 1. Le joueur choisit de jeter une carte dans la fosse.
 2. Le système jette la carte sélectionnée.
 3. Le joueur pioche une nouvelle carte.
 4. Le coup est enregistré.
- **Extensions** : N/A
- **Contraintes** : N/A