TP5 Pokedeck Mode Swing

Sommaire

[Remarque 2](#_Toc412500815)

[Application 2](#_Toc412500816)

[Modelisation 3](#_Toc412500817)

[Les choses à améliorer et à faire (auto critique) 4](#_Toc412500818)

[Bilan et difficultés 4](#_Toc412500819)

# Remarque

Le package « pokedeck » venant du projet TP3-4 précédent n’a pas été modifié ou améliorer. Je l’utilise pour les classes déjà codées. Je l’ai modifié juste pour mettre le chemin des images en tant que propriété de carte et initialiser en dure le pokedeck pour avoir un exemple de liste. Les packages à prendre en compte sont ceux du TP5.

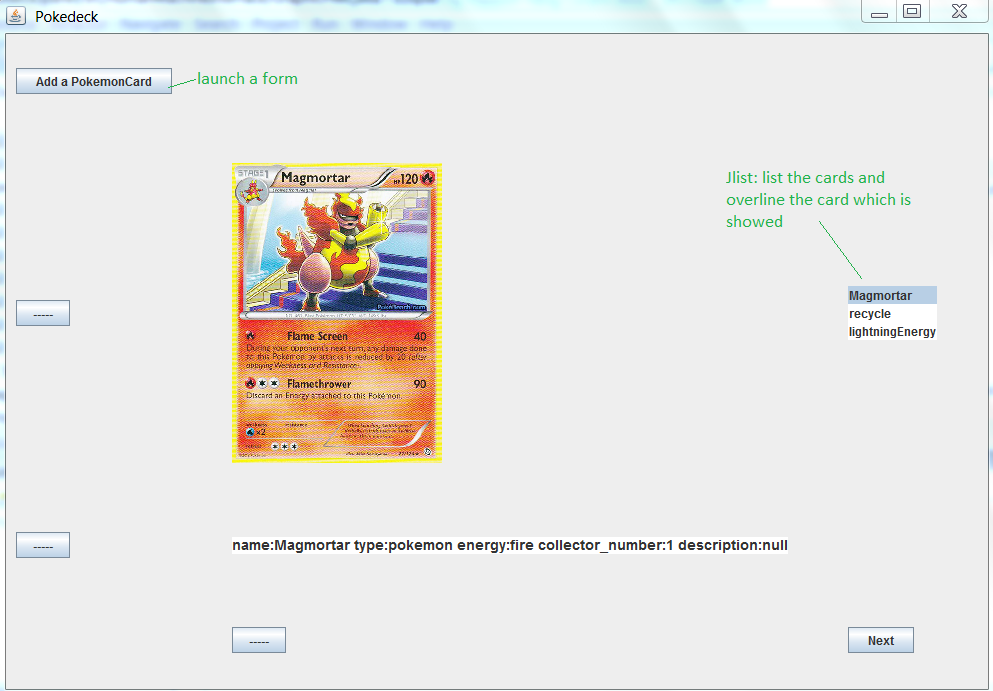
# Application

Le Pokedeck mode Swing, c’est le projet pokedeck avec une interface graphique utilisant la librairie Swing de java. L’interface développée offre la possibilité de consulter les cartes enregistrées dans le pokedeck. On peut voir la liste des cartes et appuyer sur un bouton next affichant une autre carte. Quand on arrive à la fin de la liste, c'est-à-dire la dernière carte stockée, l’application nous renvoie à la première carte (donc pas de bouton previous dans ce projet).

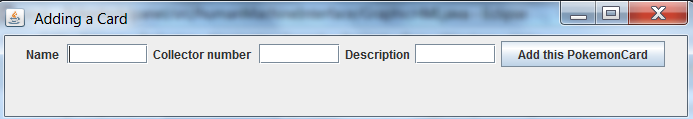
Une Jlist (qui nous montre les noms des cartes) est mis à disposition. Elle surligne le nom correspondant à la carte courante affichée.

Un formulaire pour ajouter une carte a été intégrer (appel via le bouton « add a pokemonCard »).

Les informations de la carte sont affichées dans un label en dessous de l’image.



Lorqu’on clique sur le bouton add a pokemonCard nous entrons dans une fenêtre avec un formulaire.

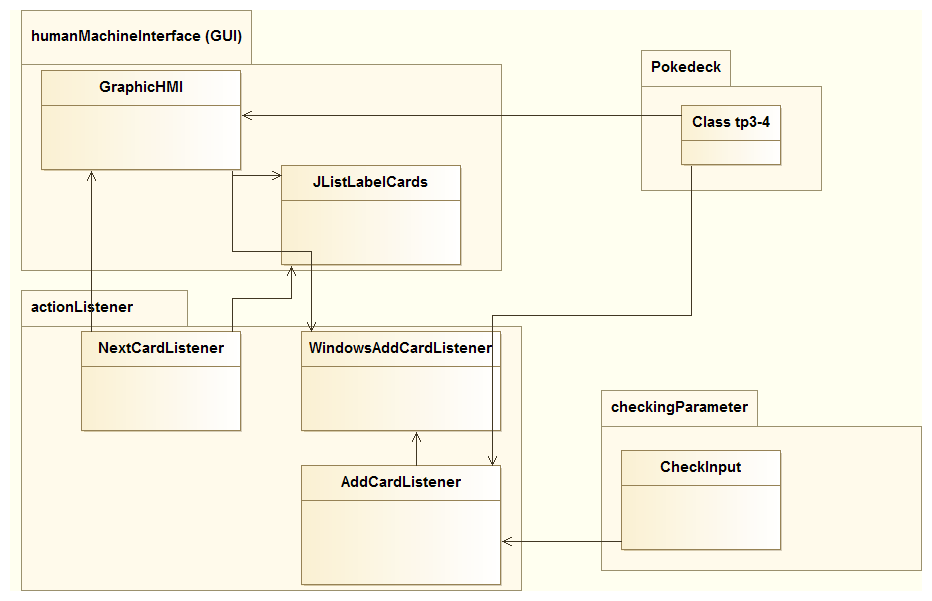


Un système de vérification a été implémenté de façon a vérifier si le Name est vide (chaine null ou ne contient que des espaces) et à détecter si le collector number est un nombre (integer). La description peut être vide Cela ajoute une carte au pokedeck si les champs sont remplis correctement.

Des ActionListener sont utilisés et le GridBagConstraint pour placer les éléments graphiques.

# Modelisation

Le schéma ne comporte que les classes issues du TP5. Les classes du TP3-4 sont représentées par le package Pokedeck.



# Les choses à améliorer et à faire (auto critique)

L’application n’est pas finie. Je n’ai pas bien défini les classes. Je devrais faire une classe par exemple pour les coordonnées quand j’utilise une GridBagConstraint.

Pour récupérer une image aussi, je devrais faire une classe spéciale juste pour l’image d’une carte.

Quand je crée un Jcomposant, la séquence d’instruction est toujours la même. Par exemple quand je crée un bouton, je devrais selon moi utiliser un design pattern pour pouvoir faire la création de composants selon leur type (bouton, label , list etc…). La structure est vraiment à revoir.

Dans la classe « WindowsAddCardListener », j’appelle des méthodes Swing pour créer une fenêtre de formulaire dans le ActionPerformed. Je dois peut être dissocié la création d’élément grapique pour le mettre dans une classe.

Je n’ai pas implémenter les autres options du menu (remove a card ,save, rechercher etc…) car je manque de temps.

L’application ne prend qu’en compte l’ajout d’une carte de type pokemon. J’ai codé en dur le type de carte. Mais je voulais rajouter une liste déroulante avec ces types (pokemon, trainer, energy). Il faut aussi rajouter une autre liste déroulante avec les sous types (fire, water ; item etc…).

# Bilan et difficultés

Je n’ai pas bien visualisé au tout début comment structurer l’application. J’ai eu du mal à voir comment séparer les fonctions modifiant les Jcomposant graphique Swing et ceux qui manipulent les données. J’ai eu aussi du mal à faire passer des informations aux classes listener. Cela m’a pris du temps car je ne sais pas vraiment comment modifier proprement l’interface graphique via un événement dynamiquement. Mais ce projet a été ludique.