Een domein specifieke taal in Scala voor Collectible Card Games – Implementatieplan

# Iteratie 1

Fase 1 van de implementatie zal 100% statisch zijn. Dit wil zeggen dat nieuwe spelelementen (kaarten, eigenschappen, regels, rondes) hard gecodeerd zijn.

## Beschikbare kaarten

* Creatures: Enkel kaarten met een aanvals- en verdedigingswaarde zijn beschikbaar. Kaarten hebben nog geen verdere eigenschappen

## Beschikbare spelfases

* Voorbereidingsfase:
  + Kaart trekken van deck [verplicht, automatisch]
  + Handkaart(en) spelen [optioneel]
* Aanvalsfase [optioneel]:
  + Aanval bekend maken [optioneel]
  + Verdediging bekend maken (door tegenstander) [optioneel]
  + Aanval en verdediging uitvoeren

## GUI functionaliteit

* Applicatieniveau:
  + Nieuw spel starten: Levenspunten herstellen, decks schudden, handen delen
  + Kaart bekijken: Alle open kaarten moeten bekeken kunnen worden
* Spelniveau:
  + Spelfase overslaan: Optionele spelfases moeten eenvoudig kunnen worden overgeslaan
  + Handkaart(en) spelen
  + Aanval bekend maken
  + Aanval uitvoeren (automatisch)
  + Verdediging bekend maken

# Iteratie 2

Fase 2 van de implementatie zal 100% statisch zijn. Dit wil zeggen dat nieuwe spelelementen (kaarten, eigenschappen, regels, rondes) hard gecodeerd zijn.

## Beschikbare kaarten

* Creatures: Kaarten kunnen al enkele eigenschappen hebben:
  + Flying: Kan alleen verdedigd worden door een creature met Flying of Reach
  + Unblockable: Kan niet verdedigd worden
  + Defender: Kan niet aanvallen
  + Reach: Kan een creature met Flying verdedigen

## Beschikbare spelfases

Geen nieuwe functionaliteit

## GUI functionaliteit

Geen nieuwe functionaliteit