Circustrein

# Requirements

FR-01 Dieren moeten worden kunnen ingevuld.

B-01.1 Er wordt ingevuld hoe groot een dier is en of het vlees eet.

K-01.1 Keuze menu’s zodat er geen foute waarden worden ingevuld.

FR-02 Dieren moeten verdeeld worden in wagons.

B-02.1 Vlees etende dieren mogen niet in een wagon met dieren van hetzelfde formaat of kleiner.

B-02.2 Klein dier is 1 punt, middelgrote dieren 3 punten en grote dieren 5 punten.

B-02.3 Alle wagons moeten optimaal benut zijn.

B-02.4 Een wagon heeft 10punten aan ruimte.

# Use-cases

|  |  |
| --- | --- |
| Naam | UC01: Dieren toevoegen |
| Samenvatting | Nadat alle gegevens van alle dieren zijn toegevoegd heeft het systeem een lijst met alle dieren. |
| Actors | Directeur Hans |
| Aannamen | geen |
| Scenario | Hans wilt zijn dieren in het systeem toevoegen, de console app vraagt hoeveel dieren hij wilt toevoegen en vraagt dan voor alle dieren de benodigde informatie. Hans vult alle gegevens in. Het systeem slaat deze gegevens op om daarna alle dieren in juiste wagons te plaatsen. |
| Uitzonderingen | Wanneer iets fout word ingevuld geeft het systeem een error en dan kan hans opnieuw de gegevens invullen. |
| Resultaat | De dieren staan in het systeem. |

|  |  |
| --- | --- |
| Naam | UC02: Dieren distribueren |
| Samenvatting | Wanneer alle dieren in het systeem staan word de beste manier om ze te verdelen berekent en is er een lijst hiervan in de console te zien. |
| Actors | Directeur Hans |
| Aannamen | Er is een lijst met dieren in het systeem |
| Scenario | Hans die heeft al een lijst met dieren ingevuld. Hij vraagt het systeem om de beste verdeling te berekenen. Het systeem berekent de beste verdeling. Het systeem laat een lijst zien met de beste verdeling. |
| Uitzonderingen | Als er geen lijst is gaat het systeem naar UC01. |
| Resultaat | Hans heeft een lijst waar die dieren het best verdeeld zijn. |

# Klassendiagram

