**Requirements document**

**Project:** dierenasiel

**Door:** Noud Habraken

**Klas:** S26



Inhoud:

[Inleiding 3](#_Toc515971422)

[MoSCoW model 4](#_Toc515971423)

[Use Cases 5](#_Toc515971424)

[Database model 9](#_Toc515971425)

[Klassen diagram 10](#_Toc515971426)

[Userinterface 12](#_Toc515971427)

[Testcases 16](#_Toc515971428)

[Traceability matrix 24](#_Toc515971429)

[Keuzes 25](#_Toc515971430)

# Inleiding

Dit project is bedoeld als zowel een administratieve tool voor de medewerkers van een dierenasiel, als wel een site voor eventuele bezoekers van het dierenasiel.

Met de winforms client kan men bijhouden wanneer een dier voor laatst eten heeft gekregen, wanneer een hond voor het laatst is uitgelaten en wanneer een hok voor het laatst is schoongemaakt. Ook kan men makkelijk dieren als gereserveerd zetten en in één oogopslag zien voor hoeveel dagen er nog eten is voor de dieren.

Met de ASP.Net website kan een bezoeker zich opgeven als vrijwilliger, kan hij/zij een dier afstaan en/of een dier reserveren. Ook kan hier informatie vinden over het dierenasiel zelf en zijn dieren.

# MoSCoW model

**Must haves:**

|  |  |
| --- | --- |
| Code: | Item: |
| M1.0 | Medewerkers moeten dieren toe kunnen voegen. |
| M1.1 | Medewerkers moeten de eigenschappen van een dier toe kunnen voegen. |
| M2 | Medewerkers moeten dieren kunnen bekijken. |
| M3 | Medewerkers moeten kunnen zien wanneer de hond voor het laatst uitgelaten is. |
| M4 | Medewerkers moeten kunnen zien wanneer het dier voor het laatst eten heeft gehad. |
| M5 | Medewerkers moeten kunnen zien wanneer het hok voor het laatst schoongemaakt is. |
| M6 | Dieren moeten gereserveerd kunnen worden. |
| M7 | Bezoekers moeten dieren kunnen kopen. |
| M8 | Bezoekers moeten dieren kunnen bekijken. |
| M9 | Nieuwe medewerkers/vrijwilligers moeten kunnen worden toegevoegd. |
| M10 | Bezoekers moeten een afspraak kunnen maken. |
| M11 | Nieuwe gebruikers moeten kunnen worden toegevoegd. |
| M12 | Medewerkers moeten in kunnen loggen. |

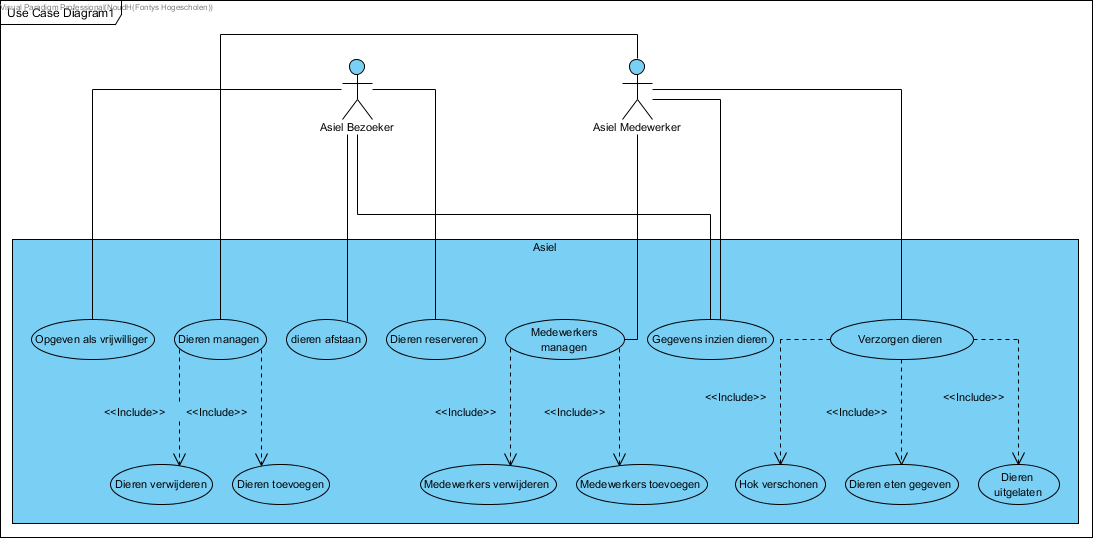
**Should haves:**

|  |  |
| --- | --- |
| Code: | Item: |
| S1 | Bezoekers moeten dieren af kunnen staan. |

**Could haves:**

|  |  |
| --- | --- |
| Code: | Item: |
| C1 | Bezoekers moeten zich op kunnen geven als vrijwilliger. |

# Use Cases

****

|  |  |
| --- | --- |
| **Naam** | Verzorgen dieren |
| **Samenvatting** | De actor kan het dier verzorgen, dat houdt in dat men het dier eten kan geven, uit kan laten en zijn/haar hok verschonen. |
| **Actor** | Asiel Medewerker |
| **Aanname** | Actor is ingelogd |
| **Beschrijving** | 1. Actor selecteert dier uit lijst van dieren. 2. Het systeem laat de mogelijk opties zien voor het verzorgen van een dier. 3. Actor kiest een verzorgingstype en voert de gegevens in. 4. Het systeem verwerkt de opdracht. |
| **Uitzondering** |  |
| **Resultaat** | Het systeem is bijgewerkt met de ingevulde informatie. |

|  |  |
| --- | --- |
| **Naam** | Gegevens inzien dieren |
| **Samenvatting** | De actor kan de gegevens van de dieren inzien. |
| **Actor** | Asiel Medewerker, Asiel bezoeker |
| **Aanname** | Actor is ingelogd |
| **Beschrijving** | 1. Actor selecteert dier uit lijst van dieren. 2. Het systeem laat de gegevens van het dier zien |
| **Uitzondering** |  |
| **Resultaat** | De actor krijgt de inzage in de gegevens van het gekozen dier. |

|  |  |
| --- | --- |
| **Naam** | Dieren afstaan. |
| **Samenvatting** | De Actor staat een gevonden dier af. |
| **Actor** | Asiel bezoeker |
| **Aanname** | Actor heeft een dier. |
| **Beschrijving** | 1. Actor vult NAW-gegevens in en informatie over het gevonden dier. 2. Het systeem verstuurt deze naar een medewerker 3. De medewerker voegt het dier toe aan het systeem. 4. Het systeem verwerkt de opdracht. |
| **Uitzondering** |  |
| **Resultaat** | Het dier is afgestaan aan het asiel en toegevoegd aan het systeem. |

|  |  |
| --- | --- |
| **Naam** | Dieren toevoegen. |
| **Samenvatting** | De Actor voegt een dier toe aan het systeem. |
| **Actor** | Asiel medewerker. |
| **Aanname** | Actor is ingelogd. |
| **Beschrijving** | 1. Actor vult informatie over het dier in. 2. Het systeem verwerkt de informatie. |
| **Uitzondering** |  |
| **Resultaat** | Het dier is toegevoegd aan het systeem. |

|  |  |
| --- | --- |
| **Naam** | Dieren verwijderen. |
| **Samenvatting** | De Actor verwijderd een dier uit het systeem. |
| **Actor** | Asiel medewerker |
| **Aanname** | Actor is ingelogd. Dier bestaat in het systeem. |
| **Beschrijving** | 1. Het systeem laat een lijst met dieren zien. 2. De Actor kiest welk dier er wordt verwijderd. [1] 3. Het systeem verwerkt de opdracht en verwijdert het dier. |
| **Uitzondering** | [1] De actor maakt geen keuze, de use case wordt afgesloten. |
| **Resultaat** | Het dier is verwijderd uit het systeem. |

|  |  |
| --- | --- |
| **Naam** | Dieren reserveren. |
| **Samenvatting** | De Actor reserveert een dier. |
| **Actor** | Asiel bezoeker |
| **Aanname** |  |
| **Beschrijving** | 1. Het systeem laat een lijst met dieren zien. 2. De Actor kiest een dier en vult zijn/haar NAW-gegevens in. [1] 3. Het systeem maakt een reservering aan. |
| **Uitzondering** | De actor vult geen NAW-gegevens in, de use case wordt afgebroken. |
| **Resultaat** | Het dier is gereserveerd. |

|  |  |
| --- | --- |
| **Naam** | Medewerker toevoegen |
| **Samenvatting** | De Actor voegt een medewerker toe aan het systeem. |
| **Actor** | Asiel medewerker. |
| **Aanname** | Actor is ingelogd. |
| **Beschrijving** | 1. Actor vult NAW-gegevens in van de nieuwe medewerker. 2. Het systeem verwerkt de opdracht. |
| **Uitzondering** |  |
| **Resultaat** | Een nieuwe medewerker is toegevoegd aan het systeem. |

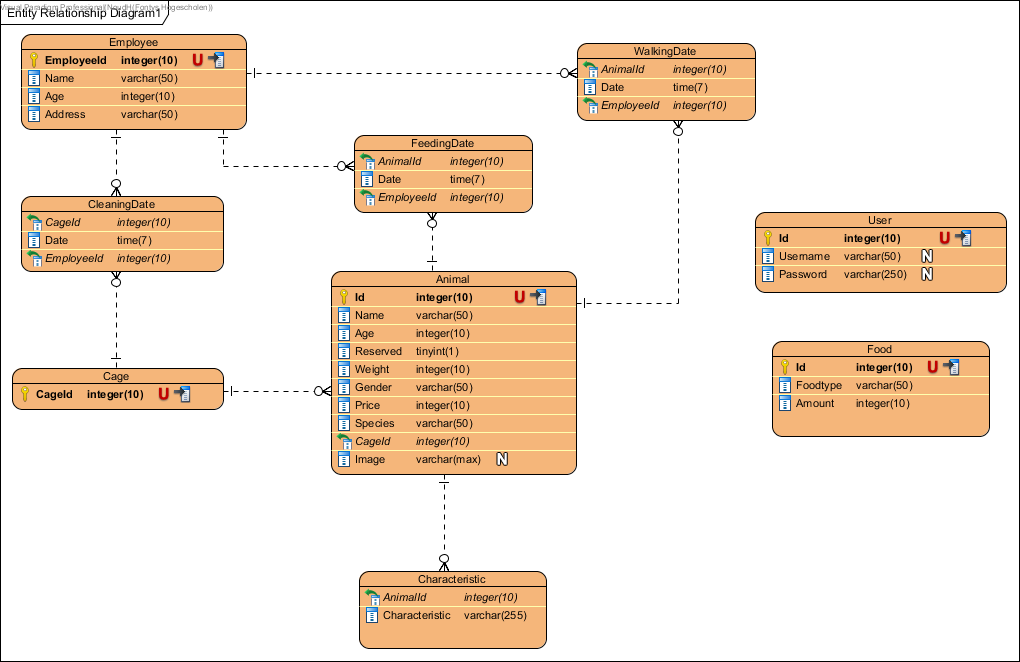
|  |  |
| --- | --- |
| **Naam** | Medewerkers verwijderen. |
| **Samenvatting** | Een medewerker wordt verwijderd uit het systeem. |
| **Actor** | Asiel medewerker |
| **Aanname** | Actor is ingelogd |
| **Beschrijving** | 1. Het systeem geeft een lijst met medewerkers weer. 2. De Actor kiest welke medewerker moet worden verwijderd. [1] 3. Het systeem verwijderd de medewerker uit de database. |
| **Uitzondering** | [1] De actor maakt geen keuze, de use case wordt afgebroken. |
| **Resultaat** | De medewerker is verwijderd uit het systeem. |

|  |  |
| --- | --- |
| **Naam** | Vrijwilliger opgeven. |
| **Samenvatting** | De Actor geeft zich op als vrijwilliger. |
| **Actor** | Asiel bezoeker |
| **Aanname** |  |
| **Beschrijving** | 1. Actor vult NAW-gegevens in. 2. Het systeem verstuurt deze naar een medewerker 3. De medewerker voegt de vrijwilliger toe aan het systeem. [1] 4. Het systeem voegt de vrijwilliger toe. |
| **Uitzondering** | De medewerker keurt de Actor af, de use case wordt afgebroken. |
| **Resultaat** | Het dier is afgestaan aan het asiel en toegevoegd aan het systeem. |

# 

# Database model

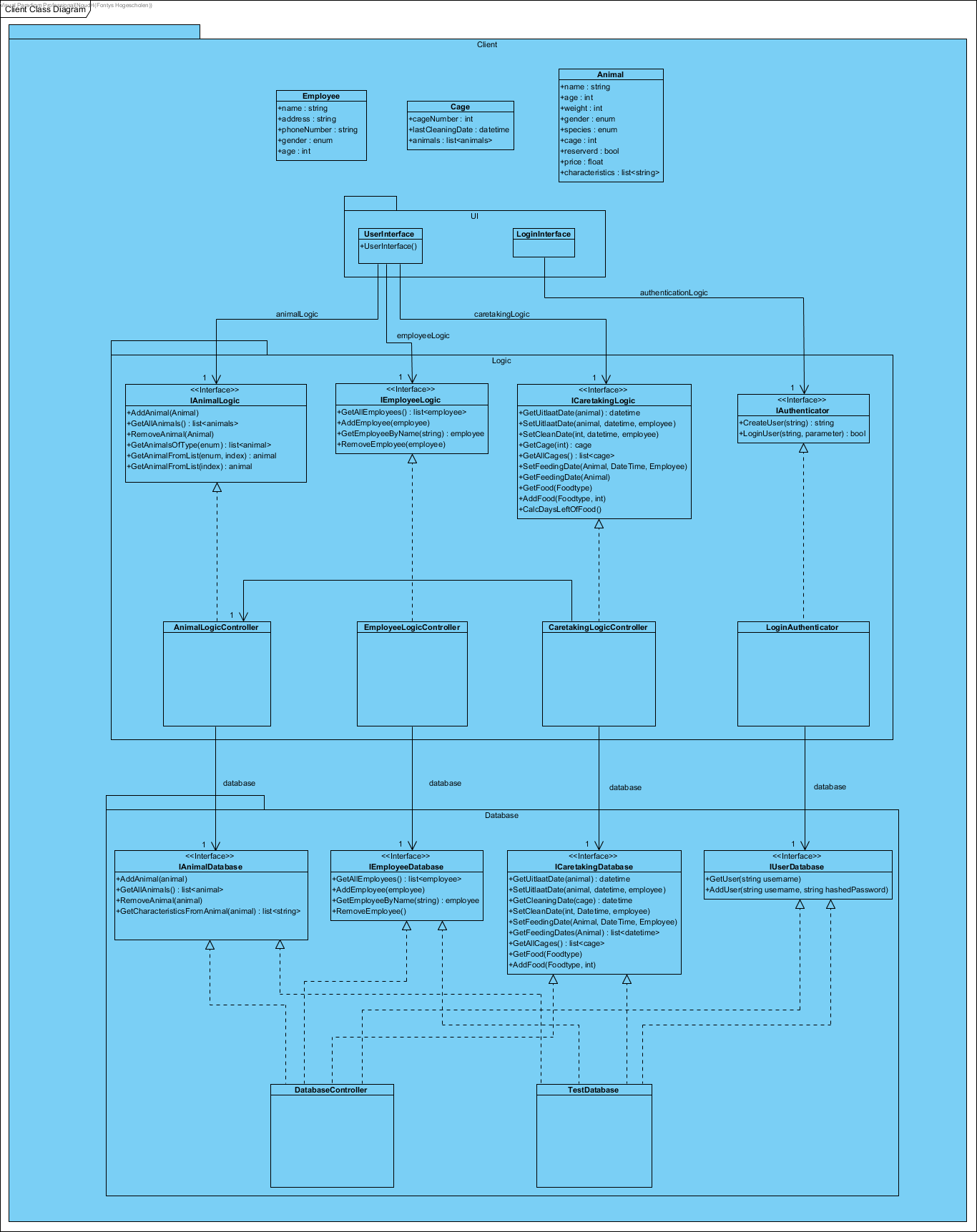
Het databasemodel laat zien hoe de database in elkaar zit.



# Klassen diagram

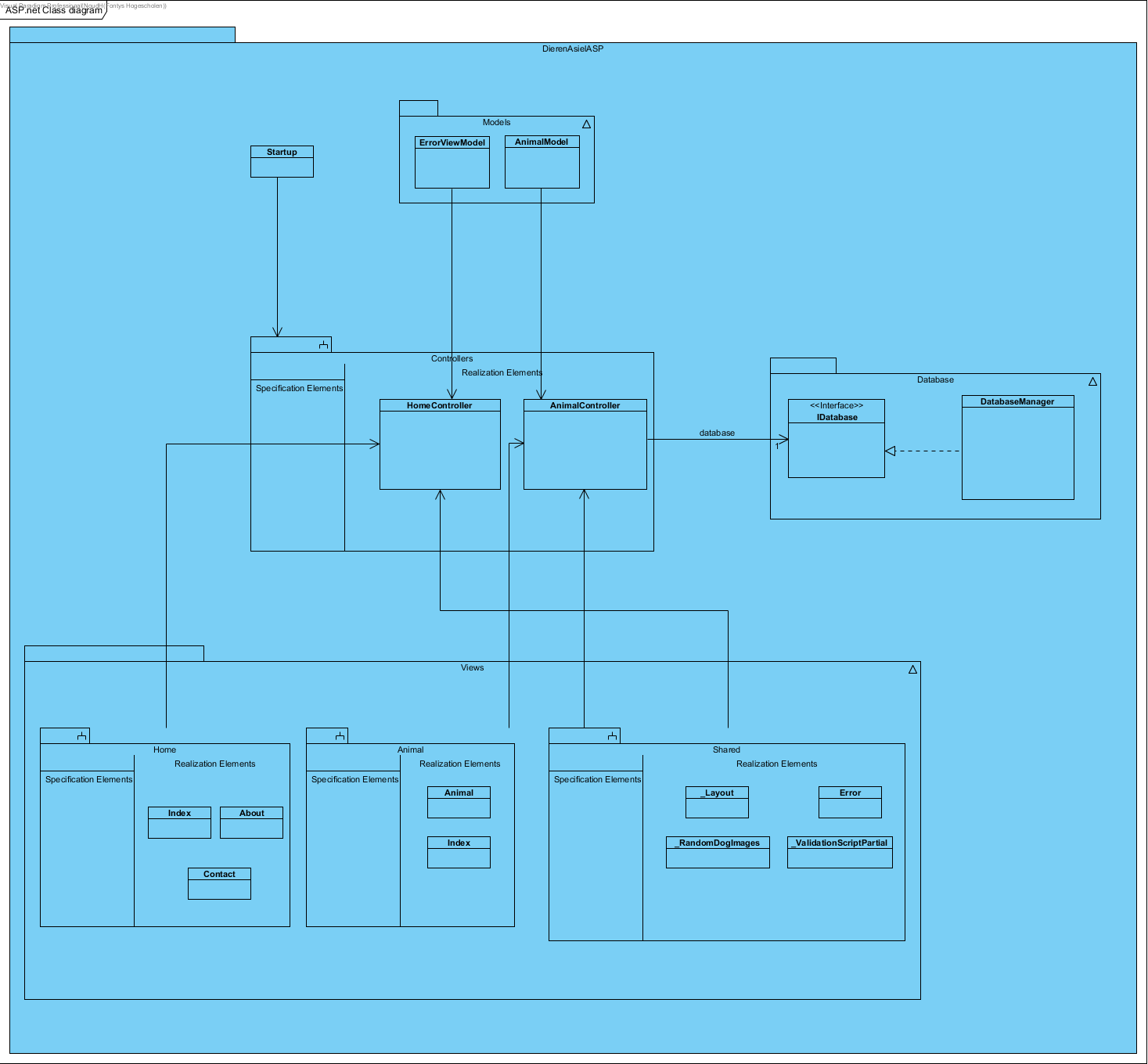
**Client:**

Dit is het klassen diagram van de winforms client. Het maakt gebruik van de drielaagsstructuur (UI, logic, database) en is bedoeld voor medewerkers als administratieve tool.



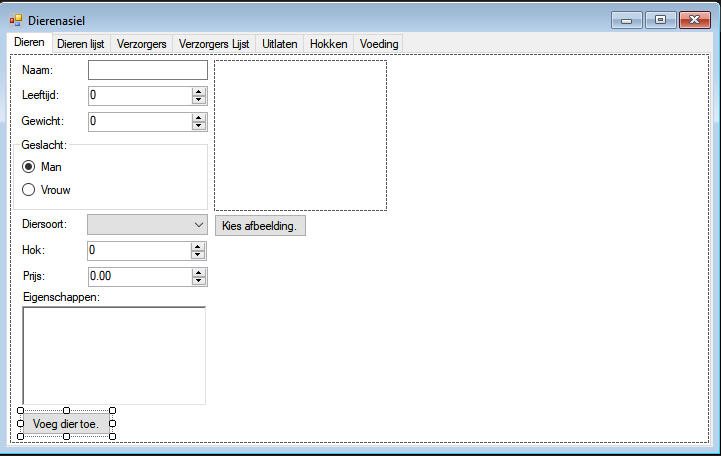
**ASP.Net**

Dit is het klassendiagram van het ASP.net project. Het vraagt de data op uit de database welke van toepassing is voor de bezoeker en laat deze vervolgens zien. Ook kan de bezoeker een dier reserveren, een dier afstaan en/of zichzelf opgeven als vrijwilliger.

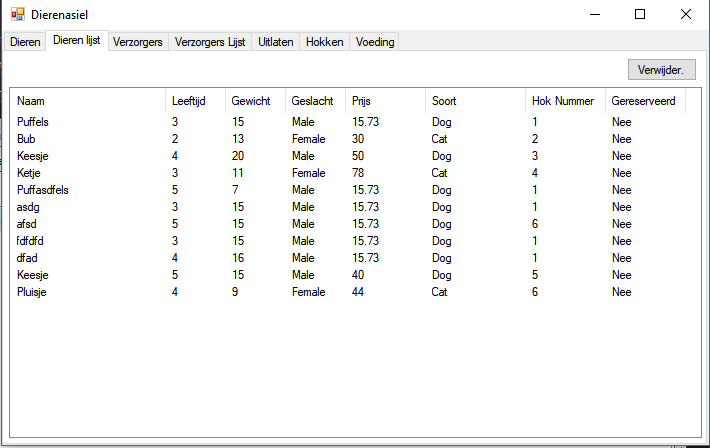
****

# Userinterface

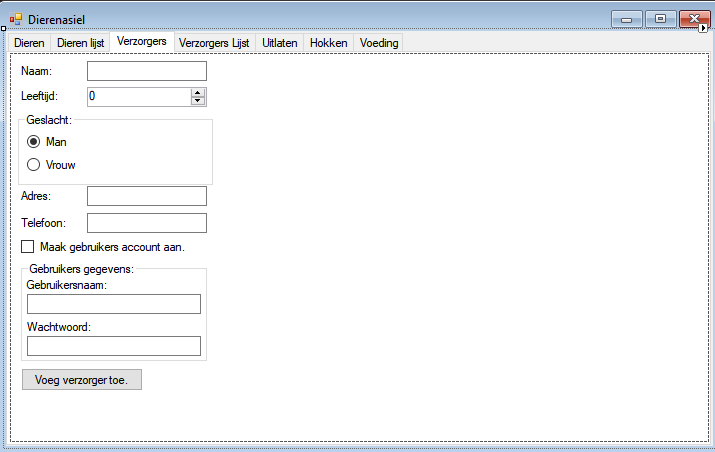
Dieren toevoegen



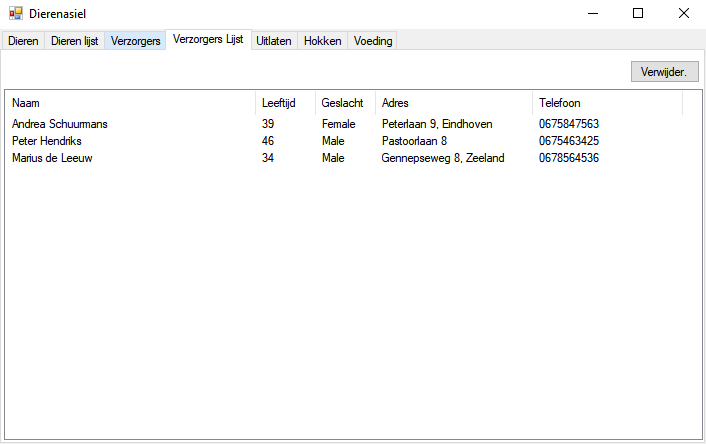
Dierenlijst



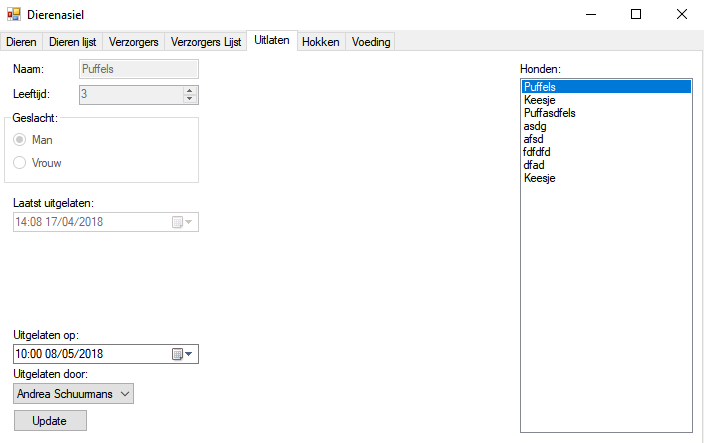
Verzorgers toevoegen



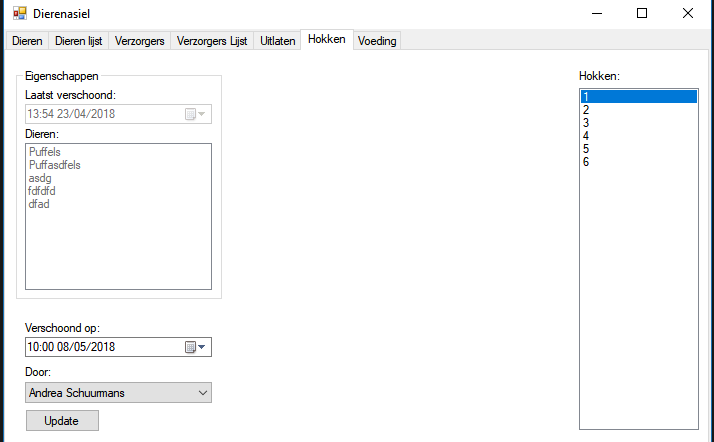
Verzorger lijst



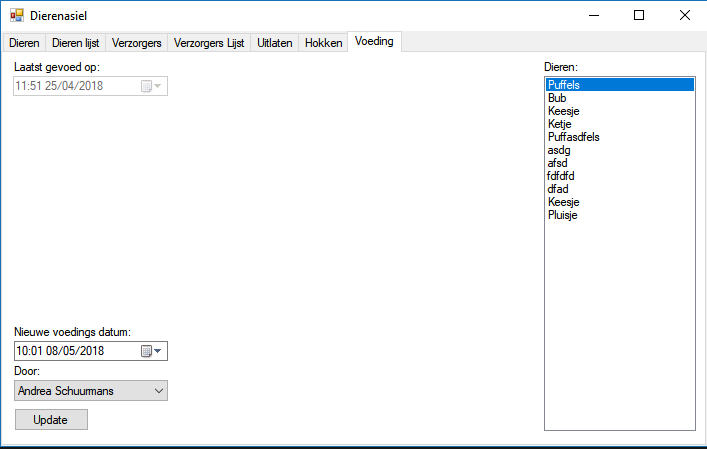
Uitlaten van honden



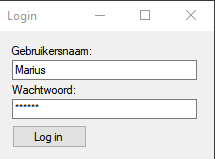
Hokken verschonen



Voeden van de dieren.



Inloggen.



# Testcases

**Inloggen**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Testcase Id | Test stappen | Verwachte resultaat | Uitkomst | Daadwerkelijke resultaat |
| Tc1.1 | Start de applicatie, er zal nu een inlog scherm verschijnen.  Login met gebruikersnaam Marius en wachtwoord Leeuw1. En klik daarna op de knop login. | De user krijgt toegang tot de applicatie. |  |  |
| Tc1.2 | Start de applicatie, er zal nu een inlog scherm verschijnen.  Login met gebruikersnaam Marius en wachtwoord leeuw1. En klik daarna op de knop login. | De user krijgt een melding te zien dat de inloggegevens incorrect zijn en dat hij deze opnieuw in moet vullen. |  |  |
| Tc1.3 | Start de applicatie, er zal nu een inlog scherm verschijnen.  Login met gebruikersnaam Piet en wachtwoord Leeuw1. En klik daarna op de knop login. | De user krijgt een melding te zien dat de inloggegevens incorrect zijn en dat hij deze opnieuw in moet vullen. |  |  |

**Dier toevoegen**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Testcase Id | Test stappen | Verwachte resultaat | Uitkomst | Daadwerkelijke resultaat |
| Tc2.1 | Start de applicatie en login, ga vervolgens naar het tabblad “Dieren”. Voer vervolgens de gegevens van een dier als volgt in: Naam: keesje;  Leeftijd: 4; gewicht: 8; dier: hond; diersoort: labrador; prijs: 250; eigenschappen: lief; plaatje: plaatje naar keuze; Verhaal: lorem ipsum;  En klik daarna op de knop “voeg toe”. | Het dier en zijn gegevens worden toegevoegd aan de database, waarna deze te zien zal zijn in de lijst met dieren. |  |  |
| Tc2.2 | Start de applicatie en login, ga vervolgens naar het tabblad “Dieren”. Laat de velden leeg en klik vervolgens op de knop voeg toe. | De user krijgt een melding te zien dat de gegevens incorrect ingevuld zijn en het dier wordt niet aan de database toegevoegd. |  |  |

**Medewerkers toevoegen**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Testcase Id | Test stappen | Verwachte resultaat | Uitkomst | Daadwerkelijke resultaat |
| Tc3.1 | Start de applicatie en login, ga vervolgens naar het tabblad “Medewerkers”. Voer vervolgens de gegevens van de medewerker als volgt in:  Naam: Sandra schuurman;  Leeftijd: 33;  Adres: molenweg 9, Veghel;  Telefoon: 0687543627  Vink de optie “Maak gebruikers account aan” aan en vul vervolgens het volgende in:  Gebruikersnaam: Sandra  Wachtwoord: Schuurman2; | De medewerker met de ingevoerde gegevens wordt toegevoegd aan het systeem en krijgt inloggegevens om in te loggen. |  |  |
| Tc3.2 | Start de applicatie en login, ga vervolgens naar het tabblad “Medewerkers”. Voer de gegevens van de medewerker als volgt in:  Naam: Peter Hendriks;  Leeftijd: 37;  Adres: paardenweg 19, Veghel;  Telefoon: 0687876747 | De medewerker met de ingevoerde gegevens wordt toegevoegd aan het systeem, maar krijgt geen inloggegevens. |  |  |
| Tc3.3 | Start de applicatie en login, ga vervolgens naar het tabblad “Medewerkers” en laat alle velden leeg. Klik vervolgens op de knop “voeg toe”. | De user krijgt een melding dat de gegevens incorrect zijn. De medewerker wordt niet aan het systeem toegevoegd en krijgt ook geen login gegevens. |  |  |

**Het uitlaten van honden**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Testcase Id | Test stappen | Verwachte resultaat | Uitkomst | Daadwerkelijke resultaat |
| Tc4.1 | Start de applicatie en login, ga vervolgens naar het tabblad “Uitlaten”.  Kies daarna een hond uit de lijst uit en voer de datum van vandaag in en kies een medewerker uit de lijst. Klik vervolgens op de knop “Update.” | Het systeem slaat de gegevens op in de database en laat de meest recente datum zien aan de user. |  |  |
| Tc4.2 | Start de applicatie en login, ga vervolgens naar het tabblad “Dieren lijst”. Klik vervolgens met de rechtermuisknop op een hond. Een knop met de tekst “uitlaten” zal verschijnen, klik hierop. Hierna zal men doorverwezen worden naar het tabblad “uitlaten” met de eerder gekozen hond als het gekozen dier uit de lijst. Kies daarna een hond uit de lijst uit en voer de datum van vandaag in en kies een medewerker uit de lijst. Klik vervolgens op de knop “Update.” | Het systeem slaat de gegevens op in de database en laat de meest recente datum zien aan de user. |  |  |

**Het eten geven van een dier**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Testcase Id | Test stappen | Verwachte resultaat | Uitkomst | Daadwerkelijke resultaat |
| Tc5.1 | Start de applicatie en login, ga vervolgens naar het tabblad “Voeding”.  Kies daarna een dier uit de lijst uit en voer de datum van vandaag in en kies een medewerker uit de lijst. Klik vervolgens op de knop “Update.” | De gegevens worden toegevoegd aan de database en laat de meest recente datum zien aan de user. |  |  |
| Tc5.2 | Start de applicatie en login, ga vervolgens naar het tabblad “Dieren lijst”. Klik vervolgens met de rechtermuisknop op een dier. Een knop met de tekst “voeden” zal verschijnen, klik hierop. Hierna zal men doorverwezen worden naar het tabblad “voeding” met de eerder gekozen dier als het gekozen dier uit de lijst.  Kies daarna een dier uit de lijst uit en voer de datum van vandaag in en kies een medewerker uit de lijst. Klik vervolgens op de knop “Update.” | De gegevens worden toegevoegd aan de database en laat de meest recente datum zien aan de user. |  |  |

**Het schoonmaken van de hokken**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Testcase Id | Test stappen | Verwachte resultaat | Uitkomst | Daadwerkelijke resultaat |
| Tc6.1 | Start de applicatie en login, ga vervolgens naar het tabblad “hokken”.  Kies daarna een hok uit de lijst uit en voer de datum van vandaag in en kies een medewerker uit de lijst. Klik vervolgens op de knop “Update.” | De gegevens worden toegevoegd aan de database en laat de meest recente datum zien aan de user. |  |  |
| Tc6.2 | Start de applicatie en login, ga vervolgens naar het tabblad “Dieren lijst”. Klik vervolgens met de rechtermuisknop op een dier. Een knop met de tekst “verschonen” zal verschijnen, klik hierop. Hierna zal men doorverwezen worden naar het tabblad “hokken” met de eerder gekozen dier zijn hok als het gekozen hok uit de lijst.  Kies daarna een dier uit de lijst uit en voer de datum van vandaag in en kies een medewerker uit de lijst. Klik vervolgens op de knop “Update.” | De gegevens worden toegevoegd aan de database en laat de meest recente datum zien aan de user. |  |  |

**Het reserveren van de dieren (Medewerkers)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Testcase Id | Test stappen | Verwachte resultaat | Uitkomst | Daadwerkelijke resultaat |
| Tc7.1 | Start de applicatie en login, ga vervolgens naar het tabblad “Dieren lijst”.  Uit de lijst met dieren wordt een dier gekozen wat nog niet gereserveerd is.  Daarna klikt men op dit dier met de rechtermuisknop, waarna een contextmenu zichtbaar wordt. Klik met de muis op “Toggle reservering”. | Het dier zal in de database bijgewerkt worden naar gereserveerd en de lijst met dieren zal de kolom “gereserveerd” op “Ja” springen. |  |  |
| Tc7.2 | Uit de lijst met dieren wordt een dier gekozen wat al gereserveerd is.  Daarna klikt men op dit dier met de rechtermuisknop, waarna een contextmenu zichtbaar wordt. Klik met de muis op “Toggle reservering”. | Het dier zal in de database bijgewerkt worden naar gereserveerd en de lijst met dieren zal de kolom “gereserveerd” op “Nee” springen. |  |  |

**Het reserveren van de dieren (gebruikers)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Testcase Id | Test stappen | Verwachte resultaat | Uitkomst | Daadwerkelijke resultaat |
| Tc8 | Op de pagina “Animals”, kies een dier en klik op de knop “Meer informatie”. Klik vervolgens op de knop “Reserveer Nu” en verstuur uw NAW-gegevens via de geopende mail. | Nadat een medewerker de mail heeft verwerkt, zal er een afspraak worden gemaakt en zal het dier gereserveerd worden. |  |  |

**Het afstaan van gevonden/ongewenste dieren.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Testcase Id | Test stappen | Verwachte resultaat | Uitkomst | Daadwerkelijke resultaat |
| Tc9 | Ga naar de website en ga vervolgens naar “Contact”. Neem vervolgens contact op met het dierenasiel en maak een afspraak. | Er zal een afspraak worden gemaakt voor het afstaan van het gevonden dier. |  |  |

**Het opgeven als vrijwilliger**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Testcase Id | Test stappen | Verwachte resultaat | Uitkomst | Daadwerkelijke resultaat |
| Tc10 | Ga naar de website en ga vervolgens naar “Contact”. Neem vervolgens contact op met het dierenasiel en maak een afspraak. | Er zal een afspraak worden gemaakt voor een gesprek met het dierenasiel. |  |  |

# Traceability matrix

Dit is de test matrix m.b.t. de requirements.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **M1.0** | **M1.1** | **M2** | **M3** | **M4** | **M5** | **M6** | **M7** | **M8** | **M9** | **M10** | **M11** | **M12** | **S1** | **C1** |
| **Tc1** |  |  |  |  |  |  |  |  |  |  |  |  | X |  |  |
| **Tc2** | X | X | X |  |  |  |  |  |  |  |  |  |  |  |  |
| **Tc3** |  |  |  |  |  |  |  |  |  | X |  | X |  |  |  |
| **Tc4** |  |  |  | X |  |  |  |  |  |  |  |  |  |  |  |
| **Tc5** |  |  |  |  | X |  |  |  |  |  |  |  |  |  |  |
| **Tc6** |  |  |  |  |  | X |  |  |  |  |  |  |  |  |  |
| **Tc7** |  |  |  |  |  |  | X |  |  |  |  |  |  |  |  |
| **Tc8** |  |  |  |  |  |  |  | X | X |  | X |  |  |  |  |
| **Tc9** |  |  |  |  |  |  |  |  |  |  |  |  |  | X |  |
| **Tc10** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | X |

# Keuzes

**Accounts voor bezoekers:**

Na wat onderzoek te doen naar andere dierenasielen en websites ben ik tot de conclusie gekomen dat accounts voor bezoekers onnodig zijn. Een dierenasiel werkt tenslotte niet zoals een winkel waar men vaak wat bestelt. Daarbij wordt informatie zoals adres, bonuspunten en/of kortingscodes opgeslagen en onthouden voor later gebruik. Dit is in het geval van een dierenasiel onnodig, het gaat tenslotte over een eenmalige aankoop van een dier.

**Mogelijkheid om als gebruiker direct te kunnen reserveren:**

Het leek mij geen goed idee om de gebruiker direct een reservering te kunnen laten maken via de site, omdat men dan alle dieren meteen zou kunnen reserveren, waardoor er vervolgens geen dieren meer worden laten zien aan bezoekers die wel echt geïnteresseerd zijn.

In plaats daarvan ben ik gekeken naar hoe andere sites dit deden. Hieruit is gekomen dat men vaak een mail sturen, waarna vervolgens een medewerker een afspraak maakt en het dier op gereserveerd zet.

**Opgeven als vrijwilliger:**

Het leek mij beter als men een mail zou sturen naar het dierenasiel. Het dierenasiel kan dan contact opnemen met de eventuele toekomstige vrijwilliger en beslissen wat hij/zij moet gaan doen.

**Opgeven gevonden/ongewenste dieren:**

Ook hier leek het mij beter als er eerst via de mail contact op wordt genomen met het dierenasiel. Deze kan dan beslissen of er ruimte is voor het dier en of het dier daadwerkelijk bestaat en het niet gewoon een grap is. Verder kan men dan daarna de gegevens van het dier invullen en het dier opnemen in het asiel.