Ontwerp Document

Live Performance

Sven Dubbeld

S21T

2835231

# **1 INHOUD**

|  |  |
| --- | --- |
| [1 INHOUD](#h.bv1y2ffcncpl) | 1 |
| [2 INLEIDING](#h.i35d2mtkls68) | 2 |
| [3 ARCHITECTUUR](#h.e3dssw3jnrn2) | 3 |
| [3.1 Gui](#h.4xvhqpwahign) | 3 |
| [3.2 Dal](#h.aj7i7g8s2f6u) | 3 |
| [3.3 Business logic](#h.ri6nvxed8dyi) | 3 |
| [4 CODE STYLE](#h.lnxbz9) | 4 |
| [4.1 Algemeen](#h.35nkun2) | 4 |
| [4.2 Indentation](#h.1ksv4uv) | 4 |
| [4.3 Naamgeving](#h.44sinio) | 4 |
| [4.4 Documentatie](#h.2jxsxqh) | 4 |
| [5 DATABASE ONTWERP](#h.z337ya) | 5 |
| [6 KLASSENDIAGRAM](#h.3j2qqm3) | 6 |

# **2 INLEIDING**

Dit document beschrijft de manier waarop de Software Engineers te werk zullen gaan aan de huur administratie van *‘t Sloepke*. Het document is opgebouwd aan de hand van de lagenstructuur.

# **3 ARCHITECTUUR**

De architectuur bestaat uit 3 lagen: de GUI, de DAL en de business logic. Het centrale framework dat gebruikt wordt is ASP.NET MVC5.

## **3.1 Gui**

In deze laag staan alle MVC controllers en views. De rol van de controllers hierin is het doorgeven van het model aan de views, die dit vervolgens aan de gebruiker presenteren.

## **3.2 Dal**

Hier staat de communicatie met de database. Deze laag bestaat uit een interface waar de applicatie mee kan communiceren en een implementatie. Deze implementatie kan verwisseld worden zonder dat de applicatie dit merkt. Hiermee kan bijvoorbeeld een In-Memory database gebruikt worden tijdens het testen. Als standaard implementatie wordt een Oracle database gebruikt.

## **3.3 Business logic**

Hier staan alle algoritmes en logica geimplementeerd. Deze laag is ook de enige die met de DAL en GUI communiceert, deze twee lagen kunnen dat onderling namelijk niet.

# **4 CODE STYLE**

## **4.1 Algemeen**

* Alle code wordt geschreven in het Engels.
* Alle commentaar wordt geschreven in het Engels.

## **4.2 Indentation**

* Indentation wordt gedaan met 4 spaties. Dit kan automatisch gedaan worden door Visual Studio.
* Accolades beginnen altijd op een nieuwe regel.

## **4.3 Naamgeving**

* **Namespace**  
  PascalCase
* **Classes**  
  PascalCase
* **Exception Classes**  
  PascalCase
* **Interfaces**  
  IPascalCase met I prefix.
* **Functions**  
  PascalCase. Geen underscores, behalve bij event handlers.
* **Properties, Public Members**  
  PascalCase
* **Parameters**  
  camelCase
* **Locale variabelen**  
  camelCase
* **Private/protected variabelen**  
  \_camelCase met \_ prefix. Altijd met protected of private in de declaration.
* **Form Controls**  
  PascalCase met het type control als prefix. Zie tabel 1.
* Alle andere variabelen krijgen geen type prefix.
* Zie ook: <https://en.wikibooks.org/wiki/C_Sharp_Programming/Naming>

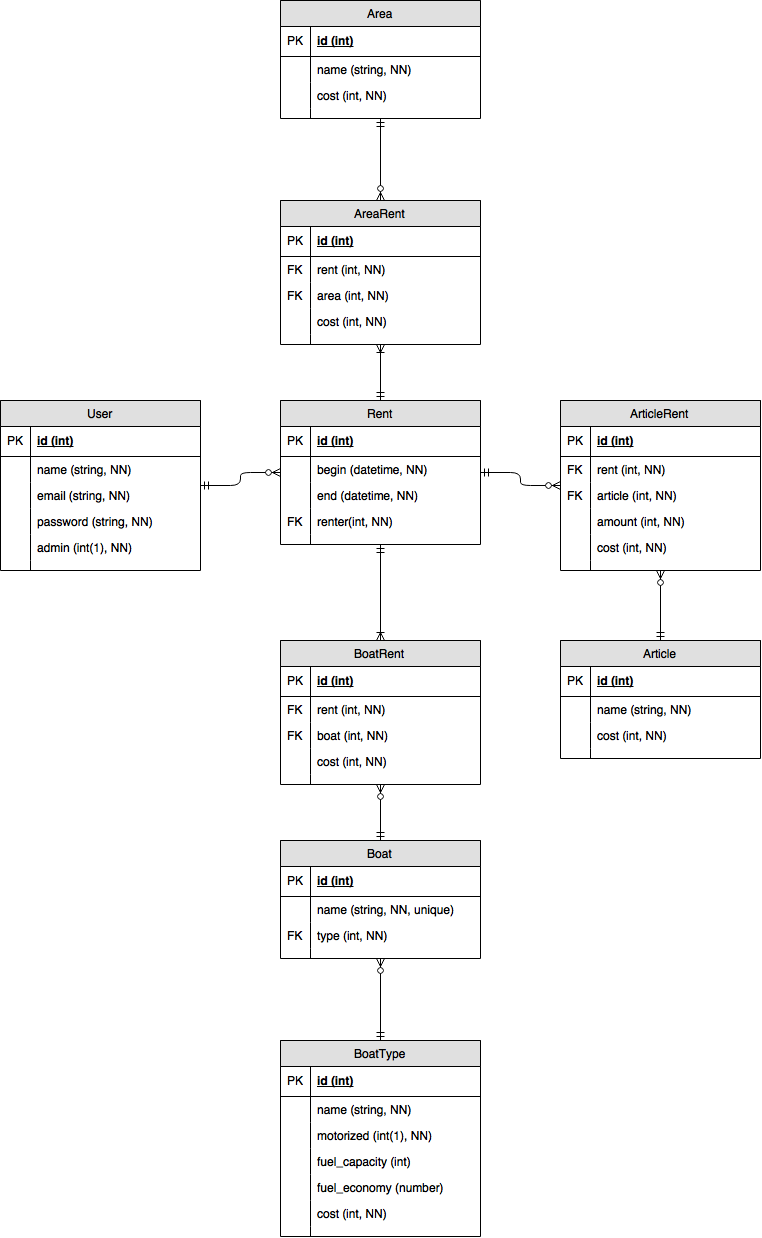
## **4.4 Documentatie**

* Single-line comments beginnen met //.
* Multi-line comments beginnen met /\* en eindigen met \*/ .

XML documentation bestaat uit single-line comments met XML tags.

# **5 DATABASE ONTWERP**

De database is de centrale plek waar alles wordt opgeslagen. Alle informatie die het systeem nodig gaat hebben is hier te vinden. De rol van de entiteiten staat beschreven in het Analyse Document in het hoofdstuk DATAMODEL.



# **6 KLASSENDIAGRAM**

Hier staat de algemene opzet van de classes die we gebruiken binnen de systemen beschreven. De domein en data laag zijn hier weergegeven.

Om het mogelijk te maken om de Data Access Layer (DAL) te kunnen swappen zonder dat de rest van de code dit merkt gebruiken we een *Injector* class om dependency injection mogelijk te maken. Hier kunnen implementaties voor een contract worden ingevoerd die later opgehaald kunnen worden, deze zijn dan voor elke class in de applicatie gelijk en kan op 1 centrale plek bepaald worden. Alle classes kunnen deze *Injector* aanroepen als static class.

De DAL heeft een *IRepository* interface voor alle tabellen. De business layer praat alleen maar tegen deze interfaces om met de database te communiceren. Deze interfaces zijn verder uitgewerkt in *OracleRepository*. Als er een andere database gebruikt moet worden dan hoeven alleen deze repositories opnieuw aangemaakt te worden en deze in te stellen als bron.

