

# **Programmeren gevorderd: Eindopdracht opgave: Fitness Reservatie**

## **1. Deadlines**

### **1.1. Deadline 1 - 18 april**

Inhoud:

- Logboek
- Klasse diagram van het domein (DCD)
- Implementatie domeinlaag
- Unit testen van de belangrijkste klassen en methodes binnen het domein
- Een repository gebaseerd op een mapper klasse (hard gecodeerde objecten)
  - deze repository wordt later vervangen door een implementatie die connecteert met een database (ADO.NET)
- Een console applicatie om eventueel functionaliteit te testen
  - deze console applicatie wordt later vervangen door een grafische applicatie (WPF)

### **1.2. Deadline 2 - 15 mei**

Inhoud:

- Logboek
- Verdere afwerking van DCD, domeinlaag en unit testen
  - Waar mogelijk gebruik makend van collections en linq
- Een repository die connecteert met een database

### **1.3. Deadline 3 - Finale oplevering: Datum nader te bepalen (eind mei of juni)**

Inhoud:

- Logboek
- Verdere afwerking van DCD, domeinlaag en unit testen
- Verdere afwerking persistentielaag

- Eén of twee grafische applicaties (klant en beheerder) die gebruik maken van de domeinlaag.
- Demo: 3 demo filmpjes met de gevraagde user stories

## 2. Fitness reservatiesysteem - Beschrijving

Het **fitness centrum** beschikt over een aantal **fitnesstoestellen** (voorlopig enkel **loopbanden** en **fietsen**). Elk van die toestellen wordt uniek geïdentificeerd door een numerieke code.

De bedoeling is dat een aangemelde **klant** een **reservatie** kan maken voor een toestel voor een bepaald tijdslot. De tijdslots die zijn voorzien zijn telkens 1 uur lang. We onderscheiden ochtendsessies tussen 8 en 12 (dus 4 slots), middagsessies vanaf 12h tot 18h en avondsessies van 18h tot 22h. In totaal zijn er dus 14 slots op 1 dag.

Voor het maken van een reservatie geef je op voor welke dag en tijdslot je een bepaald toestel wenst te reserveren. Er zijn echter voorwaarden verbonden aan een reservatie, zo kan je één bepaald toestel maximaal 2 slots na elkaar reserveren en je kan maximaal 4 slots reserveren op 1 dag.

Een reservatie bevat de volgende info :

- klantnummer, emailadres, voor en achternaam
- datum
- voor elk tijdslot het gereserveerde toestel

Zorg er ook voor dat er enkel reservaties kunnen plaatsvinden in de toekomst (lijkt me evident) en maximaal 1 week op voorhand.

Enkel toegelaten voor een **beheerder**: Het kan voorkomen dat een toestel stuk gaat en er onderhoud nodig is, dan moet dat toestel tijdelijk niet beschikbaar worden. Er kunnen ook nieuwe toestellen geleverd worden en die moeten dan ook aan de lijst met te reserveren toestellen worden toegevoegd. Als we een toestel verwijderen uit de fitness dan moeten wel nog alle reservaties uit het verleden beschikbaar blijven. We verwijderen trouwens geen toestellen die nog reservaties hebben.



- Unieke codes voor de toestellen moeten door de databank worden aangemaakt.
- Ook reservatienummers moeten door het systeem worden aangemaakt.

Beschikbare gegevens:

- Er is een lijst van klanten beschikbaar met daarin de volgende info (klant\_nr, voornaam, achternaam, email, adres, geboortedatum, interesses en klanttype – Gold, Silver, Bronze).
  - Deze lijst zal later beschikbaar gemaakt worden. In eerste instantie kan je zelf enkele klanten voorzien.
- Er is ook een lijst met toestellen beschikbaar (toestelnummer en toesteltype)
  - Deze lijst zal later beschikbaar gemaakt worden. In eerste instantie kan je zelf enkele toestellen voorzien.

Persistentielaag:

Volgende zaken worden bewaard in de persistentielaag: fitnessstoestellen, klanten en reservaties. Voor elk zal dus een repository nodig zijn met de nodige functionaliteit.

User stories:

- Als klant identificeer ik mij in het systeem op basis van mijn klantnummer of emailadres
- Als klant kan ik een reservatie aanmaken voor een bepaalde dag en tijd, daarbij geef ik aan welk toestel ik wens te gebruiken in welk tijdslot.
- Als beheerder kan ik een toestel toevoegen aan het systeem.
- Als beheerder kan ik een toestel verwijderen uit het systeem.
- Als beheerder kan ik een toestel tijdelijk buiten gebruik zetten.

## 3. Opdracht



Maak een ontwerp voor de domeinlaag. Gebruik hiervoor Visual Paradigm.



Maak de nodige software aan gebruik makend van een 3-lagen architectuur.

Voor de UI (let op: dit geldt enkel voor de laatste deadline): focus op gebruiksvriendelijkheid (keuzelijstjes, checkbox, ... maak het de gebruiker zo gemakkelijk mogelijk) en zorg er ook voor dat de schermen schaalbaar zijn.

Focus bij je ontwerp op onderhoudbaarheid / uitbreidbaarheid / efficiëntie.

Voorzie de nodige kwaliteitsgaranties door het maken van Unit Tests.



Begin aan deze opdracht vanuit het standpunt van een beheerder.

### 3.1. Te gebruiken technologie

- Visual Studio
- Visual paradigm (voor het DCD)
- C#
- .NET 5.0
- Unit testen
- SQL server
- ADO.NET
- WPF

## 4. Finale oplevering en verdediging - Demo

Maak 3 demo filmpjes met de volgende user stories :

- User story : als klant maak je een reservatie op voor een middagtraining waarbij je een fietstoestel reserveert voor 2 uren en aansluitend een loopband voor 1 uur.
- User story : als fitnesszaal-beheerder voeg je een loopband toe aan de lijst met beschikbare toestellen.
- User story : als fitnesszaal-beheerder plaats je een toestel (fiets) in onderhoud.