**Alfa College**

**Project Forum**

**Project 1**

**Technisch Ontwerp**

*In opdracht van* : J. Koster

*Auteur(s)* :

*Klas* :

*Datum* :

*Versienummer* :

*Plaats* :

Projectleider :

**Inhoud**

[1 Inleiding 4](#_Toc451973978)

[2 Klassendiagram 5](#_Toc451973979)

[3 Genormaliseerde gegevensverzameling 6](#_Toc451973980)

[3.1 Normalisatie studierooster 6](#_Toc451973981)

[3.2 Normalisatie inschrijflijst 7](#_Toc451973982)

[3.3 Normalisatie Rooster 7](#_Toc451973983)

[3.4 Samenvoeging normalisaties 8](#_Toc451973984)

[4 ERD 9](#_Toc451973985)

*<<<<<-------------------------- Deze bladzijden verwijderen ----------------------------->>>>>*

***Het Technisch Ontwerp***

*In het Technisch Ontwerp maak je de bouwtekening van het informatiesysteem dat je gaat realiseren. Je legt uit, aan de hand van het Functioneel Ontwerp (FO) en het Grafisch Ontwerp (GO) welke technische oplossingen je hebt gekozen. Werk alle details volledig uit. Benoem bijvoorbeeld de keuzes voor software, de programmeertalen waar je gebruik van gaat maken, de keuze voor een specifiek CMS enz. Vermeld ook waarom je die keuzes hebt gemaakt. Vervolgens leg je uit hoe een collega ApplicatieOntwikkelaar het project moet gaan uitvoeren. Je instructies moeten zo duidelijk zijn dat iemand anders het informatiesysteem zonder problemen kan realiseren, zonder dat hij of zij vragen hoeft te stellen.*

*Een belangrijk onderdeel van het TO is de gegevensstructuur. Aan de hand van de onderdelen die van belang zijn voor de applicatie met hun kenmerken en de contacten die er bestaan tussen die onderdelen stel je een* ***klassendiagram*** *op. Deze klassen zijn dezelfde klassen die we in het Object-Oriented programmeren kennen.*

*Houd er zoals bij elk verslag rekening mee, voor wie je het schrijft. Dit verslag schrijf je voor degene die het project gaat realiseren en voor de projectleider. Dat betekent dat de technische delen voldoende details bevatten, zodat degene die de volgende stap(pen) gaat uitvoeren hier alle benodigde informatie uit kan halen.*

*Taalgebruik: Zakelijk*

*Doelgroep: afhankelijk van het soort test, dus wel of geen vaktaal.*

*<<<<<-------------------------- Deze bladzijden verwijderen ----------------------------->>>>>*

# Inleiding

*In de inleiding geef je kort aan waar dit project over gaat en wat de inhoud is van dit document.*

*De hele inleiding is maximaal één alinea (ca. 8 regels).*

Dit document is het Technisch Ontwerp voor het project Studentenregistratiesysteem ASTE Universiteit. In dit TO wordent allereerst het klassendiagram weergegeven dat is opgesteld aan de hand van het FO. Het hoofdstuk Genormaliseerde Gegevensverzameling laat de normalisaties per rapportage zien met aan het einde de samengevoegde normalisatie. Tenslotte kunt in in het hoofdstuk ERD het uitgewerkte EntiteitenRelatieDiagram vinden.

# Klassendiagram

*Een klassendiagram is een weergave van de objecten en hun relaties. Een object bestaat uit de volgende drie onderdelen.*

* *De naam van een object. Deze is uniek binnen het diagram.*
* *De attributen of eigenschappen van een object*
* *De operaties of methoden. De handelingen die verricht moeten worden op of door het object.*

*Het klassendiagram is een vertaling van de functionaliteiten uit het FO. Zorg er dus voor dat je ze allemaal verwerkt en niets vergeet.*

Hieronder volgt het klassendiagram van de ASTE Universiteit.



# Genormaliseerde gegevensverzameling

*De genormaliseerde gegevensverzameling is gebaseerd op de rapportages van de opdrachtgever. Voor elk van de rapportages wordt een normalisatie uitgevoerd van de nulde tot en met de derde normaalvorm. Het eindresultaat is dus een set gegevensverzamelingen in de derde normaalvorm. Deze worden met behulp van synoniemen en homoniemen op een juiste wijze samengevoegd.*

## Normalisatie studierooster

*De normalisatie voldoet aan de volgende eisen.*

* *De normalisatie begint met een lijst gegevens waarin onderscheid wordt gemaakt tussen elementaire gegevens, procesgegevens en constante gegevens. Dit wordt ook wel de inventarisatie of nulde normaalvorm (0NV) genoemd.*
* *Zet een dubbele streep onder een sleutel (primary key)*
* *Zet een enkele streep onder een vreemde sleutel (foreign key)*
* *Bij elke normaalvorm wordt het gehele product van normaliseren genoteerd. Dus ook de entiteiten die niet veranderen, worden meegenomen naar de volgende normaalvorm.*
* *Bij de laatste normaalvorm (dus als er niet meer verder genormaliseerd kan worden) dienen de entiteiten van een functioneel juiste naam te worden voorzien. Deze naam dient een zelfstandig naamwoord in enkelvoud te zijn.*
* *Er dient te worden genormaliseerd tot en met de derde normaalvorm. Indien de derde normaalvorm gelijk is aan de tweede normaalvorm, moet dit expliciet vermeld worden.*
* *Voor elke rapportage vindt een aparte normalisatie plaats.*

**Lijst gegevens**

Titel vast gegeven

Datum procesgegeven

Student\_id elementair gegeven

Student samengesteld gegeven

Student\_voornaam elementair gegeven

Student\_achternaam elementair gegeven

Vakcode elementair gegeven

Categorie elementair gegeven

Docentcode elementair gegeven

Lesuur elementair gegeven

Niveau elementair gegeven

Gemiddeld startniveau procesgegeven

Gemiddeld eindniveau procesgegeven

Totaal lesuren procesgegeven

**0e normaalvorm**

**student\_id**, student\_voornaam, student\_achternaam, RG (vakcode, categorie, docentcode, lesuur, niveau)

**1e normaalvorm**

**student\_id**, student\_voornaam, student\_achternaam

**student\_id**, **vakcode**, categorie, docentcode, lesuur, niveau

**2e normaalvorm**

**student\_id**, student\_voornaam, student\_achternaam

**student\_id**, **vakcode**, lesuur

**vakcode**, categorie, docentcode, niveau

**3e normaalvorm**

Student **student\_id**, student\_voornaam, student\_achternaam

Rooster **student\_id**, **vakcode**, lesuur

Vak **vakcode**, categorie, docentcode, niveau

Docent **docentcode**

Categorie **categorie**

## Normalisatie inschrijflijst

**Lijst gegevens**

Titel vast gegeven

Datum procesgegeven

Student\_id elementair gegeven

Student elementair gegeven

Vak elementair gegeven

Categorie elementair gegeven

Docentcode elementair gegeven

Docentnaam elementair gegeven

Telefoon elementair gegeven

Totaal inschrijvingen procesgegeven

**0e normaalvorm**

**student\_id**, student RG (vakcode, categorie, docentcode, telefoon)

**1e normaalvorm**

**student\_id**, student

**student\_id**, **vakcode**, categorie, docentcode, docentnaam, telefoon

**2e normaalvorm**

**student\_id**, student

**student\_id**, **vakcode**

**vakcode**, categorie, docentcode, docentnaam, telefoon

**3e normaalvorm**

Student **student\_id**, student

StudentVak **student\_id**, **vakcode**

Vak **vakcode**, categorie, docentcode

Docent **docentcode**, docentnaam, telefoon

Categorie **categorie**

## Normalisatie Rooster

**Lijst gegevens**

Titel vast gegeven

Datum procesgegeven

Student elementair gegeven

Vak elementair gegeven

Docent elementair gegeven

Lesuur elementair gegeven

Startniveau elementair gegeven

Eindniveau elementair gegeven

**0e normaalvorm**

**student\_id**, student RG (vakcode, docent, lesuur, startniveau, eindniveau)

**1e normaalvorm**

**student\_id**, student

**student\_id**, **vakcode**, docent, lesuur, startniveau, eindniveau

**2e normaalvorm**

**student\_id**, student

**student\_id**, **vakcode**, lesuur

**vakcode**, docent, startniveau, eindniveau

**3e normaalvorm**

Student **student\_id**, student

Rooster **student\_id**, **vakcode**, lesuur

Vak **vakcode**, docent, startniveau, eindniveau

Docent **docent**

## Samenvoeging normalisaties

*Uit de samenvoeging van alle derde normaalvormen ontstaan de uiteindelijke derde normaalvormen op basis waarvan het Entiteit Relatie Diagram gemaakt wordt.*

**3e normaalvorm studierooster**

Student **student\_id**, student\_voornaam, student\_achternaam

Rooster **student\_id**, **vakcode**, lesuur

Vak **vakcode**, categorie, docentcode, niveau

Docent **docentcode**

Categorie **categorie**

**3e normaalvorm inschrijflijst**

Student **student\_id**, student

StudentVak **student\_id**, **vakcode**

Vak **vakcode**, categorie, docentcode

Docent **docentcode**, docentnaam, telefoon

Categorie **categorie**

**3e normaalvorm rooster**

Student **student\_id**, student

Rooster **student\_id**, **vakcode**, lesuur

Vak **vakcode**, docent, startniveau, eindniveau

Docent **docent**

**3e normaalvorm samenvoeging**Student **student\_id**, student\_voornaam, student\_achternaam

Rooster **student\_id**, **vakcode**, lesuur

Vak **vakcode**, categorie, docentcode, startniveau, eindniveau

Docent **docentcode**, docentnaam, telefoon

Categorie **categorie**

# ERD

*Het EntiteitenRelatieDiagram is de grafische weergave van de samengevoegde normalisaties. Het bestaat uit twee onderdelen.*

* *Entiteiten: een entiteit heeft altijd een naam en eigenschappen (attributen), overgenomen uit de normalisatie.*
* *Relaties tussen entiteiten: deze relaties geven weer welke relatie er tussen twee entiteiten is. Een relatie is altijd een-op-een, een-op-nul, een of meer, of een-op-meer.*

*Het is dus belangrijk dat de onderdelen uit de samengevoegde normalisaties zorgvuldig worden overgenomen in het ERD. Zorg dat je de juiste relaties tussen de juiste entiteiten aangeeft. Zorg er ook voor dat je geen attributen vergeet.*

*Vervolgens vul je dit ERD aan met de gegevens uit het klassendiagram. Het kan geen kwaad om het klassendiagram even te controleren aan de hand van het FO.*