**Afbeelding met tekst

Automatisch gegenereerde beschrijving**

**Studiehandleiding**

**SOFWARE LANGUAGE 01**

WFADSD.SL01.23

**Studiehandleiding WF**

Datum: 05-04-2023

Van: Polina Kozlova & Stephan Hoeksema

Versie: 0.1  
Status: Ontwikkeling

Aantal pagina’s:??

# Inleiding Software Language 01

In de module "Software Language 01" wordt geleerd hoe een softwareapplicatie ontwikkeld kan worden door middel van het gebruik van een project definitie en diverse UML en ERD-diagrammen als hulpmiddelen.

Het belang van modelleren met behulp van UML en ERD-diagrammen en de relatie tussen modelleren en programmeren wordt niet altijd goed begrepen door programmeurs. Om het toepassen en begrip te vergroten gaan we praktische voorbeelden gebruiken om de studenten te leren hoe problemen kunnen worden geanalyseerd en oplossingen kunnen worden gevonden met behulp van UML en ERD-diagrammen.

Daarnaast wordt er geleerd hoe deze diagrammen kunnen worden omgezet naar de programmeertaal en het maken van een database, met behulp van object-georiënteerd programmeren en moderne database software.

De student toont zijn praktische kennis aan door middel van een opdracht die deel uitmaakt van de module. De opdracht zal al voorzien zijn van de UML en ERD-diagrammen.

# Algemene informatie

**Gegevens van de onderwijseenheid:**

|  |  |
| --- | --- |
| **Titel onderwijseenheid** | Software Language 01 |
| **Code onderwijseenheid (VOE)** | WFADSD.SL01.23 |
| **Hoort bij onderwijsperiode** | 1 |
| **Module verantwoordelijke** | Stephan Hoeksema  s.hoeksema@windesheim.nl |
| **Docent(en) Module** | Stephan Hoeksema  Polina Kozlova |
| **School** | Hogeschool Windesheim in Almere |
| **Opleiding** | AD Software Development |
| **Niveau** | Propedeuse |
| **Leerlijn** | ICT |
| **Studiepunten (EC’S)** | 5 |

|  |
| --- |
| **Samenhang tussen het onderwijs** |
| Body of knowledge and Skills: |
| Front-end programmeren richt zich op de ontwikkeling van de gebruikersinterface en het zichtbare deel van een applicatie, terwijl back-end programmeren zich richt op de ontwikkeling van de niet-zichtbare delen van een applicatie, zoals de server-side logica, de database en de API's.  Deze twee aspecten van softwareontwikkeling werken samen om een volledige applicatie te maken. De front-end code wordt uitgevoerd op de computer of het apparaat van de gebruiker en communiceert met de back-end code die op de server draait. De back-end code verwerkt de gegevens van de gebruiker, voert berekeningen uit en communiceert met andere systemen, en geeft vervolgens de resultaten weer aan de front-end van de applicatie.  Om een goed werkende applicatie te maken, moeten front-end en back-end goed op elkaar zijn afgestemd en met elkaar communiceren. Dit kan worden bereikt door middel van een MVC model, waarmee gegevens tussen de front-end en de back-end kunnen worden uitgewisseld. |

|  |  |
| --- | --- |
| **Activiteit** | **Studiebelasting (SBU)** |
| **Uitvoeren praktijkopdracht** | 7x Hoorcollege  7x Werkcollege |
| **Bijeenkomsten** | 14 uur hoorcollege, 24 uur werkcollege uur |
| **Zelfstudie, voorbereiding, etc.** | 102 uur |
| **Totaal** | **140 uur (5\*28)** |

NB. 1 ECTS komt overeen met 28 studiebelastingsuren.

Inhoudsopgave

[Inleiding Software Language 01 2](#_Toc144212683)

[Algemene informatie 3](#_Toc144212684)

[Opleidingscompetenties 6](#_Toc144212685)

[Leersituatie en inhoud 7](#_Toc144212686)

[Doelstelling: 7](#_Toc144212687)

[Onderwijsinhoud 8](#_Toc144212688)

[Hoorcollege 01 – PHP 9](#_Toc144212689)

[Hoorcollege 02 – PHP 9](#_Toc144212690)

[Hoorcollege 03 – PHP 9](#_Toc144212691)

[Hoorcollege 04 – PHP 10](#_Toc144212692)

[Hoorcollege 05 & 6 – PHP 10](#_Toc144212693)

[Hoorcollege 07 – PHP 10](#_Toc144212694)

# Opleidingscompetenties

Deze module richt zich op de ontwikkeling van de volgende beroepstaken uit de HBO-i domeinbeschrijving, architectuurlagen: Gebruikersinterface, Organisatieprocessen, Software, en optioneel Infrastructuur (versie 2018), op niveau 1. Klik op de link voor extra inhoudelijke toelichting (HBO-I).

Hieronder vindt u de uitwerking van de architectuurlaag Software:

* Analyseren: In deze activiteit wordt functionele informatie verzameld en gevalideerd voor een softwaresysteem met één belanghebbende, volgens een standaardmethode. Daarnaast worden acceptatiecriteria voor deze functionele eisen gedefinieerd.
* Adviseren: Hierbij worden aanbevelingen gedaan over specifieke requirements van een softwaresysteem op basis van onderzoek naar bestaande, vergelijkbare systemen.
* Ontwerpen: Deze activiteit omvat het maken van een ontwerp voor een softwaresysteem, inclusief database, met behulp van modelleertechnieken volgens een standaardmethode.
* Realiseren: In deze activiteit wordt een eenvoudig softwaresysteem gebouwd, getest en beschikbaar gesteld. Het opzetten, vullen en bevragen van een database maakt deel uit van het softwaresysteem.
* Manage & Control: Hierbij wordt de softwareontwikkeling ondersteund door het inrichten van Azure DevOps omgeving en het gebruik van een beheersysteem, in teamverband.

De module stelt studenten in staat om praktische vaardigheden te ontwikkelen op het gebied van softwareontwikkeling, in overeenstemming met de HBO-i domeinbeschrijving. Niveau 1 van de beroepstaken wordt behandeld en er wordt gewerkt aan het verzamelen van functionele eisen, ontwerpen van software, bouwen van een eenvoudig systeem, en het beheren van de ontwikkeling in teamverband.

# **Leersituatie en inhoud**

Doelstelling:

Het doel voor studenten bij het volgen van de module "Software Language" is om een grondige basis te leggen in het begrijpen, ontwikkelen en effectief toepassen van softwareprogrammering. Tijdens de module worden studenten bekend gemaakt met belangrijke concepten, tools en technieken die essentieel zijn voor softwareontwikkeling. De module heeft als hoofddoel studenten voor te bereiden op het effectief oplossen van complexe problemen door middel van programmeeroplossingen, waarbij de nadruk ligt op het begrijpen van probleemdefinities, het ontwerpen van oplossingen en het implementeren van deze oplossingen met behulp van programmeertalen en tools.

Concreet worden de volgende doelen nagestreefd:

1. **Begrip van Fundamentele Concepten**: Studenten moeten begrip ontwikkelen van basisconcepten zoals variabelen, datatypes, condities, lussen, functies en objectgeoriënteerde programmering.
2. **Analyse en Ontwerp**: Studenten moeten in staat zijn om problemen te analyseren en oplossingen te ontwerpen met behulp van modelleringstechnieken zoals UML en ERD-diagrammen. Hierdoor kunnen ze de structuur en logica van softwareoplossingen plannen.
3. **Programmeerpraktijk**: Studenten moeten bekend raken met een programmeertaal (bijvoorbeeld PHP) en deze effectief kunnen gebruiken om oplossingen te implementeren. Ze moeten ook leren hoe ze code kunnen schrijven, begrijpen en debuggen.
4. **Database-integratie**: Studenten moeten leren hoe ze databases kunnen ontwerpen, beheren en gebruiken in combinatie met softwareoplossingen. Dit omvat het begrijpen van SQL-query's, databaseontwerp en basisprincipes van gegevensbeheer.
5. **Samenwerking en Teamwerk**: De module kan mogelijk ook de nadruk leggen op samenwerking en teamwerk, waarbij studenten leren om in een teamverband te werken aan softwareprojecten, versiebeheer te begrijpen en code efficiënt te delen.
6. **Praktische Toepassing**: Studenten moeten de geleerde concepten en vaardigheden kunnen toepassen op praktische projecten, waarbij ze problemen omzetten in functionele softwareoplossingen.
7. **Kritisch Denken**: De module beoogt studenten te stimuleren om kritisch te denken en oplossingen te evalueren op basis van efficiëntie, onderhoudbaarheid en schaalbaarheid.

Over het algemeen is het doel van de module "Software Language" om studenten een sterke basis te bieden in softwareontwikkeling, hen in staat te stellen om te begrijpen hoe software werkt, hoe ze het kunnen maken en hoe ze het kunnen aanpassen aan verschillende problemen en behoeften.

Aan het einde van de module:

* De student begrijpt de basisprincipes van het programmeren en past deze toe, zoals variabelen, functies, classes, loops en conditionele statements.
* De student gebruikt deze principes om gegevens uit een database te halen en deze op een webpagina te tonen.
* De student maakt een eenvoudige, interactieve website met een back-endapplicatie in PHP.
* De student lost problemen in de backend-applicatie op met behulp van de juiste methodieken.

Onderwijsinhoud

PHP staat voor Hypertext Preprocessor en het is een programmeertaal die veel wordt gebruikt voor het maken van dynamische webpagina's.

In deze module leert de student de basisprincipes van PHP, zoals variabelen, functies, loops en conditionele statements. De student leert ook hoe je PHP kunt gebruiken om gegevens uit een database te halen en deze op een webpagina te tonen.

Programmeren in PHP is een belangrijke vaardigheid voor webontwikkelaars en kan helpen om dynamische en interactieve websites te maken. Het is ook een goede basis voor het leren van andere programmeertalen, zoals JavaScript, C# en Python.

Deze module bestaat niet alleen uit theorie, maar er zal ook veel tijd besteed worden aan de praktische toepassingen! Er zal dan ook een eenvoudige, interactieve website gemaakt worden. Het zelf kunnen oplossen van fouten in deze applicatie komt ook aan bod.

Hoorcollege 01 – PHP

Doelstellingen:

* GIT – IDEA
* Comments
* Variables
* Conditionals

Hoorcollege 02 – PHP

Doelstellingen: Array's - LOOPS

Deze week staat in het teken van array's, een krachtige datastructuur in PHP. Je leert hoe je array's kunt maken, waarden kunt toevoegen en ophalen, en verschillende bewerkingen kunt uitvoeren.

Loops, waarmee je herhaalde taken kunt uitvoeren. Je leert het gebruik van verschillende soorten loops, zoals for, while en foreach.

Hoorcollege 03 – PHP

Doelstellingen: **Functions - require/include**

Functies stellen je in staat om stukken code te hergebruiken. Je leert hoe je functies kunt definiëren, parameters kunt doorgeven, waarden kunt retourneren en hoe je functies effectief kunt gebruiken om je code modulair en onderhoudsvriendelijk te maken.

Je leert hoe je code kunt scheiden en modulariteit kunt verbeteren door functies en classes uit externe bestanden in te sluiten. Dit stelt je in staat om je code beter te organiseren en herbruikbaarheid te vergroten.

Hoorcollege 04 – PHP

Doelstellingen: **Routing - PDO**

Model-View-Controller (MVC) ontwerppatroon en het werken met het PHP Data Objects (PDO) voor database-interactie.

Hoorcollege 05 & 6 – PHP

Doelstellingen: **MySQL**

We gaan aan de slag met MySQL, een populaire relationele database. Je leert hoe je verbinding maakt met een MySQL-database vanuit PHP, query's uitvoert om gegevens op te halen en te wijzigen, en hoe je je PHP-applicatie kunt integreren met een database om gegevens op te slaan.

Hoorcollege 07 – PHP

Doelstellingen: **Refactor**

Refactoren van code verwijst naar het proces van het herstructureren en verbeteren van bestaande code zonder de externe functionaliteit ervan te wijzigen. Het is het proces van het herschrijven van code om het leesbaarder, efficiënter en onderhoudbaarder te maken, zonder dat de beoogde werking veranderd.