



**Vlaamse Dienst voor Arbeidsbemiddeling en Beroepsopleiding**

## **PHP ADVANCED**

### **Herkansingstest “Kassasysteem”**



## 1 GEGEVENS VAN DE CURSIST

Naam:	
Datum:	
Start:	
Einde:	

## 2 GEGEVENS VAN DE EINDOPDRACHT

### 2.1 Verwachte voorkennis

De module PHP Advanced volledig doorgenomen hebben.

### 2.2 Doelstellingen van de eindopdracht

Deze eindoefening controleert je kennis van de verschillende aspecten die in de cursus PHP Advanced aan bod zijn gekomen. Er wordt nagegaan of je in staat bent de verschillende aangeleerde technieken te combineren in één groter project:

- Ontwerp van een databank
- Opmaak van een gebruikersinterface
- Toepassen van het MVC model in je code
- Foutafhandeling gebruiken

Ook de werkwijze en analytisch inzicht worden beoordeeld aan de hand van deze onderwerpen

### 2.3 Afspraken

Voorziene duurtijd:	1 week	
Materiaal dat je mag gebruiken:	Cursus:	<input checked="" type="checkbox"/>
	Eigen nota's:	<input checked="" type="checkbox"/>
	Handboeken:	<input checked="" type="checkbox"/>
	Internet:	<input checked="" type="checkbox"/>
Benodigdheden:	Software:	IDE naar keuze
	Hardware:	/
	Bestanden:	

### 3 OPDRACHT

Je gaat een kassasysteem ontwikkelen voor een bistro (men serveert er dranken en maaltijden).

#### Deel 1 : Kelners

Dit kassasysteem zal voornamelijk gebruikt worden door de **kelners** die in de zaak werken om **bestellingen** van klanten te registreren. Elke kelner heeft een uniek nummer (geheel getal), en zal dit moeten ingeven bij het begin van een bestelling.

Tijdens het ingeven van de bestelling maakt de kelner gebruik van **pictogrammen** die overeenkomen met de maaltijden / dranken. Uiteraard moet er op elk moment een overzicht zijn van de bestelling én moet die aangepast kunnen worden (de kelner kan zich vergist hebben in het gekozen product).

Producten zijn onderverdeeld in **categoriën** en subcategoriën (maximum 2 niveaus). Ze worden ook volgens deze hiërarchie op het scherm getoond.

- Dranken
  - o Koude dranken
  - o Warme dranken
  - o Alcoholische dranken
- Maaltijden
  - o Voorgerechten
  - o Vleesgerechten
  - o Visgerechten
  - o Desserten

Bestellingen zijn steeds gelinkt aan een **tafelnummer**. De gegevens van een tafel zijn dus als volgt :

- Tafelnummer
- Maximum aantal personen
- Locatie van de tafel (bv aan het raam, op het terras, enz..)

Er kunnen meerdere bestellingen per tafel aangemaakt worden.

Bij elke bestelling én **bestellingslijn** kan een kleine opmerking voorzien worden (bv geen citroen in de cola).

Voorzie de mogelijkheid voor een kelner om de totale rekening af te drukken wanneer de klant erom vraagt (bv bij het verlaten van de zaak). Op dit moment worden de bestellingen gewijzigd van “actief” naar “betaald”. Ze mogen niet verwijderd worden, want de zaakvoerder wil op het einde van de dag een overzicht van alle inkomsten kunnen opvragen, alsook een inventaris van de bestelde dranken/maaltijden (om de voorraad te kunnen aanvullen).

Indien de kelner zich vergist heeft dan kan een bestelling ook geannuleerd worden.

## Deel 2 : Zaakvoerder

Voorzie ook een interface voor de zaakvoerder mét login, zonder registratie<sup>1</sup>. Op elk moment moet het mogelijk zijn om volgende gegevens op te vragen (steeds per lopende dag) :

- Totale omzet (= betaalde bestellingen)
- Overzicht van bestelde producten (bv 6 cola, 15 pils, 2 spaghetti, enz.)
- Omzet uitgesplitst per kelner (betaalde versus actieve bestellingen)

Wees creatief inzake de front-end (voorzie een eigen naam en logo voor de bistro), maar zorg ervoor dat het intuïtief is.

De code moet geschreven worden volgens het MVC model.

Maak gebruik van namespaces.

**Maak eerst het datamodel en bespreek dit met je instructeur vooraleer je begint te coderen.**

---

<sup>1</sup> De zaakvoerder heeft een eigen login, een registratiemogelijkheid moet *niét* voorzien worden. Maak het *wél* mogelijk om het wachtwoord te resetten indien hij dit vergeten is.