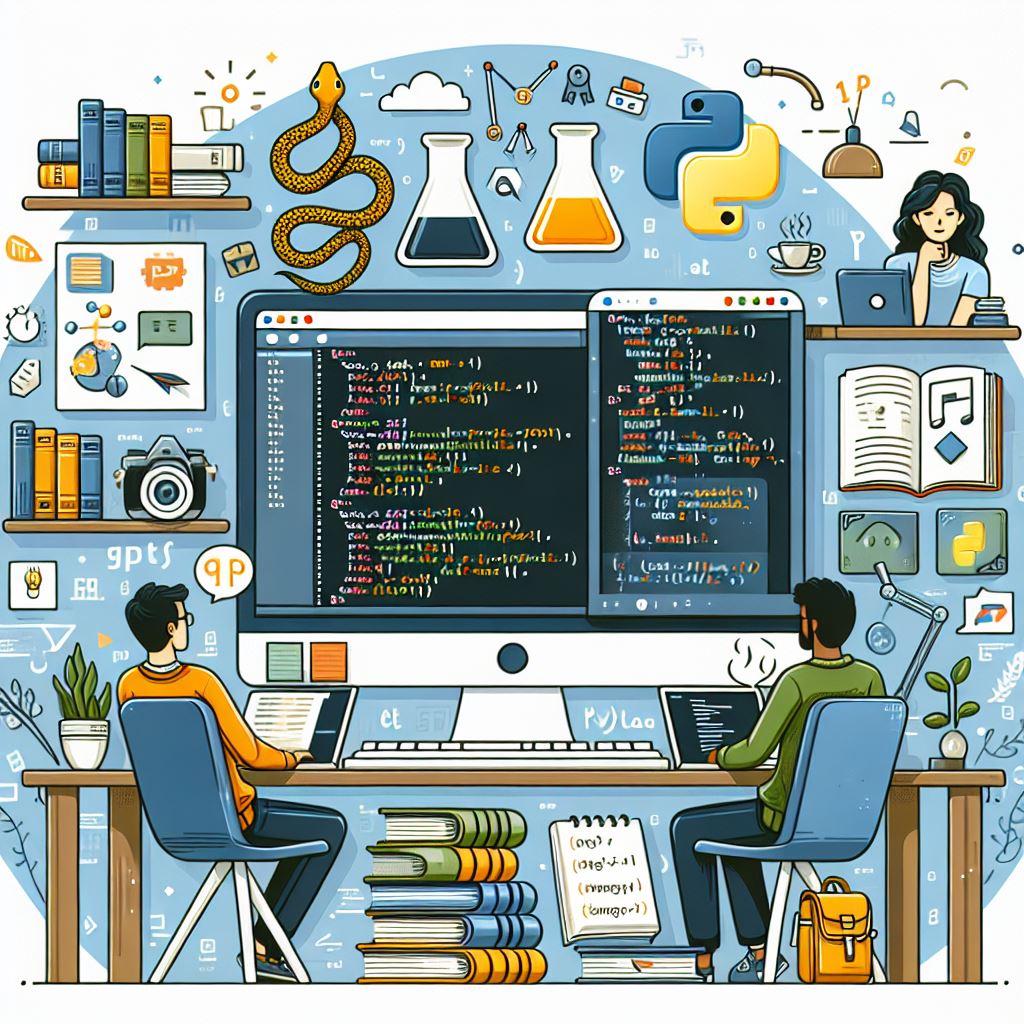
Eindopdracht

DiP2: Python Flask + Databases



Alfa-college Boumaboulevard

Team ICT

Hoite Prins

Wouter Weges

Inhoudsopgave

Inleiding 1

Startpunt 2

Downloaden startpunt 2

Initialiseren 3

Opdracht 4

Stroomschema 5

Verplichte functionaliteiten 6

Login functionaliteit (10 punten)\* 6

Logging (10 punten)\* 6

API (10 punten)\* 7

CRUD (10 punten)\* 7

Keuzefunctionaliteiten 8

Keuze 1: Reserveren (15 punten) 8

Keuze 2: Externe API gebruiken (15 punten) 8

Bonus-functionaliteiten 9

Inleveren 10

Beoordeling 11

# Inleiding

Vaardigheidstoets DiP2

In deze toets evalueer je jouw vaardigheden op het gebied van programmeren en databases.

Je krijgt een startpunt, waarop je functionaliteiten gaat toevoegen.

**Benodigdheden**

* [Git](https://git-scm.com/downloads)
* Een IDE zoals bijvoorbeeld [Visual Studio Code](https://code.visualstudio.com/download)
* Python(3) && pip(3)

# Startpunt

## Downloaden startpunt

Maak een nieuwe map in je documentenmap, open deze map in een terminal.

Download het startpunt met het volgende commando:

git clone <https://github.com/AlfaCollegeTeamICT/DIP2_eindopdracht.git>

# Initialiseren

Open je zojuist geclonede startpunt in jouw IDE.

Maak een venv aan en installeer de requirements.

python -m venv .venv  
pip install -r requirements.txt

# Opdracht

Je werkt bij AlfaFlix, een hippe retro videotheek waar je dvd’s kunt huren. Je baas wil echter met de tijd meegaan en heeft jou gevraagd om een applicatie te ontwikkelen die zowel door klanten als medewerkers kan worden gebruikt.

Er is al een basis opgezet, maar je baas komt er niet verder mee. Het is aan jou om deze wensen om te zetten in nieuwe functionaliteit!

Daarnaast zorg je ervoor dat de pagina’s er uniform en aantrekkelijk uitzien, gebruik hiervoor bijvoorbeeld Bootstrap.

Maak ook gebruik van '[Template Inheritance](https://www.digitalocean.com/community/tutorials/how-to-use-templates-in-a-flask-application#step-2-using-template-inheritance)' om je HTML-templates overzichtelijk en herbruikbaar te maken.

## Stroomschema

**Stroomschema tekenen\* (voorwaardelijk)**

Voor de functionaliteiten die je toevoegt, moet je een stroomschema maken. Je opdrachtgever wil minimaal één stroomschema zien.

* Kies een van de hieronder genoemde functionaliteiten en bouw een stroomschema van maximaal 20 stappen.
* Gebruik de juiste symbolen en labels.
* Schrijf een uitlegdocument waarin je elke stap van het schema toelicht.

## Verplichte functionaliteiten

### Login functionaliteit (10 punten)\*

Maak een registreer- en inlogfunctionaliteit die voldoet aan de volgende eisen:

* + Inloggen met gebruikersnaam & wachtwoord
  + Registreren van een nieuwe gebruiker
  + Uitloggen
  + Een “videotheek” pagina die alleen toegankelijk is voor ingelogde gebruikers

### Logging (10 punten)\*

Als een dvd wordt gehuurd, teruggebracht of gereserveerd (indien gekozen), moet er een nieuwe regel aan de logging-tabel worden toegevoegd met de volgende informatie:

* + Actie: Terugbrengen, huren (of reserveren)
  + Wie voert de actie uit?
  + Timestamp van de actie

API (10 punten)\*

Maak een API waarmee acties via de website kunnen worden uitgevoerd. Zorg voor documentatie door middel van Swagger.

* + CRUD-acties via de API (Create, Read, Update, Delete)
  + Swagger-documentatie via een swagger.yaml bestand

CRUD (10 punten)\*

Maak zichtbaar welke dvd’s beschikbaar, uitgeleend of gereserveerd (keuzeopdracht) zijn.

* + Acties: dvd’s lenen en terugbrengen
  + Overzicht van alle dvd’s (uitgeleend en beschikbaar)

## Keuzefunctionaliteiten

Je moet minstens één van de onderstaande functionaliteiten kiezen, maar je mag ook beide uitvoeren voor extra punten.

Keuze 1: Reserveren (15 punten)  
Soms willen klanten een dvd reserveren om deze later op te halen.

* Voeg een reserveringsfunctie toe aan de website.
* Voeg een reserveringsactie toe aan de website
* Reserveren werkt alleen voor ingelogde gebruikers
* Reserveren moet ook via de API mogelijk zijn
* Gereserveerde dvd’s kunnen niet worden uitgeleend

### Keuze 2: Externe API gebruiken (15 punten)

Gebruik een externe API om aanvullende informatie over de films te tonen. Dit kan op een aparte pagina of geïntegreerd worden in de videotheekpagina.

* Haal een omschrijving en een afbeelding van de film op via een externe API
* Toon de omschrijving en afbeelding bij de desbetreffende film

## Bonus-functionaliteiten

Bedenk zelf een functionaliteit en voeg deze toe. Het aantal punten wordt bepaald op basis van onder andere technische complexiteit, originaliteit, en de kwaliteit van de uitvoering. Hier zijn enkele suggesties voor mogelijke functionaliteiten:

* Voeg 2FA (twee-factor-authenticatie) toe om de beveiliging van de inlogfunctionaliteit te verbeteren.
* Implementeer gebruikersrollen:
  + Minimaal twee rollen: **admin** en **gebruiker**
  + Maak een admin-paneel waar de **C**reate en **D**elete acties van CRUD kunnen worden uitgevoerd
* Voeg OAuth-authenticatie toe, zodat gebruikers kunnen inloggen met een extern account (bijv. Google of Facebook)

# Inleveren

1. Vul het “assessment.md” bestand in en geef aan welke functionaliteiten jij gemaakt hebt.
2. Maak een **private** github-repository en voeg hier beide docenten aan toe:
   1. hj.prins@alfa-college.nl
   2. wg.weges@alfa-college.nl
3. “push” je gemaakte werk naar de repository
4. Lever een link naar de repository in op ItsLearning

* Voor hulp met Github, zie deze [video](https://youtu.be/VNTAYKDA-AE).

# Beoordeling

Het gemaakte werk wordt beoordeeld door Hoite Prins & Wouter Weges.

Het resultaat wordt gepubliceerd via Eduarte binnen tien (werk)dagen na oplevering vanaf de deadline.