

# Project WFFLIX

## Het leerplatform voor code

Auteur: Alex van der Steen, team E2

M.b.v. teamleden E2: Lorenzo Colin, Mert Kirez, Jake Zweers, Diamé Morreau

Versie: 4

Datum: 02-10-2020

Dit projectrapport is opgesteld n.a.v. de opleiding Associate Degree Software Development van Hogeschool Windesheim te Almere. Uitgevoerd door team E2 van dezelfde opleiding.

Het doel van dit projectrapport is het analyseren, systematiseren en configureren van het project WFFLIX. Een video- en educatieplatform die dient om het automatiseren en optimaliseren van distributiekkanalen te bevorderen en om mogelijk nieuwe markten te vinden.

Voor dit project gaan we werken met de Scrum methode om flexibiliteit en optimalisatie te bevorderen.

### Inhoudsopgave

Aanleiding projectorganisatie	3
Stakeholders	3
Productvisie	3
Projectgrenzen	3
Randvoorwaarden	4
Vraagstelling	4
Probleemstelling	4
Doelstellingen	5
Use cases	5-7
ERD Diagram	8
Activiteit diagrammen en testacties	9-12



## Aanleiding projectorganisatie

WFFLIX is een initiatief van de opleiding ADSD van hogeschool Windesheim Almere. Het opzet van het platform is om distributiekkanalen voor educatie beter te maken. Hiermee kan men gebruikers verwerven die mogelijk geïnteresseerd zijn in code.

Het is dus essentieel voor de opleiding ADSD dat dit product de belangen van Windesheim Almere vervult. Daarbij is team E2 van ADSD verantwoordelijk voor het leveren van het uiteindelijk gerealiseerde video- en leerplatform doorheen de sprints waardoorheen het werk verricht wordt.

## Stakeholders

Wie meedoen aan het project is belangrijk, omdat deze mensen invloed kunnen uitoefenen over het project:

- Product Owners: ○ Rudy Borgstede ○ Matthijs de Vos ○ Stephan Hoeksema
- Het projectteam E2, die de uitvoering van het project verzorgt.

## Productvisie

Met WFFLIX is het onze visie om een website te maken waar men met gemak tot kennis komt van codetalen. Deze website bevat functionaliteiten die toepasselijk zijn voor een video- en leerplatform. Het product bestaat uit een gebruikersgericht platform waarmee inloggen en registreren belangrijke functies zijn. Om video's te kunnen zien is het ook een vereiste om een account te hebben.

## Projectgrenzen

Het project loopt vanaf de kick-off in week 37 tot week 44 (begin van de toetsenweken)

Dit is dus van 07-09-2020 tot en met 25-10-2020

Het is niet de bedoeling dat wij het volgende doen:

- Het toevoegen van zelfgemaakte video's aan de website
- Het in de toekomst beheren en onderhouden van WFFLIX
- Het werven van klanten voor WFFLIX
- Onderzoeken naar de marktwaaarde van het uiteindelijke product

## Randvoorwaarden

Wat is er minimaal nodig voor de uitvoering van het project:

1. Een webserver met PHP 7.3+.
2. Een MySQL database waarop men gebruikers en video's kan beheren.
3. Een eigen ontwerp als huisstijl.

## Vraagstelling

Wat we gaan maken is een website voor videomateriaal die voldoet aan het volgende:

1. Een toegankelijke inlogfunctie.
2. Een duidelijke navigatiebalk om de gebruiker op weg te helpen op de site.
3. Voldoende mogelijkheden aan interactiviteit met de gebruiker.
4. Systemen voor bereikbaarheid zoals een contactpagina.
5. De gebruiker alleen te laten zien wat voor hem/haar relevant is. Hiermee bevorderen we het gevoel van vrijwilligheid in de gebruiker op de site.

## Probleemstelling

In onze maatschappij is het belang aan heldere en waardevolle informatieoverdracht groot bij mensen die nieuw zijn in een bepaald subject. Mensen met een codeerinteresse kunnen afgeschrikt worden door services die geld kosten, te moeilijk zijn of zelfs ontoegankelijk voor de amateur programmeur. Het kan ook zijn dat misinformatie en opzettelijk vage uitleg de gebruikers weerhoudt van nuttige informatieoverdracht.

## Doelstellingen

Het is dus ons doel om deze mensen op onze site op te vangen waarmee ze niet worden geconfronteerd met misinformatie, 'gatekeeping' en vragen die niet beantwoord kunnen worden. Door toegankelijkheid en motivatie voor de gebruiker voorop te stellen geven we deze een stimulerende omgeving die bijdraagt aan waardevolle informatieoverdracht. De accountprogressie vermeld in de vraagstelling (punt 5) staat ons ook toe om gevorderde gebruikers aanbiedingen te geven voor de ADSD van Windesheim.

## Use cases

Use case tabel voor het videofiltersysteem voor de klant.

<b>Versie nummer</b>	1.0
<b>Use case naam</b>	Video zoeken
<b>actoren</b>	klant
<b>doel</b>	Systeem juiste video aan klant laten zien
<b>triggers</b>	Bestaand klant typt naam van juiste course
<b>Pre-conditie</b>	Klant is aangemeld
<b>Post-conditie</b>	Klant kan video course bekijken Systeem weergeeft juiste video voor klant
<b>hoofdscenario</b>	1.actor drukt op filter balk 2.actor typt de video course in die hij wilt zien 3.systeem slaat ingevoerde gegevens in 4.systeem laat ingetypte video course zien aan actor zien
<b>Alternatieve scenario's</b>	geen
<b>excepties</b>	1.1 actor sluit browser 2.1 actor keert terug naar hoofdscherm 3.1 actor meld zich af
<b>kwaliteiten</b>	Bij enige vorm van excepties het systeem laten terug keren in vorige staat

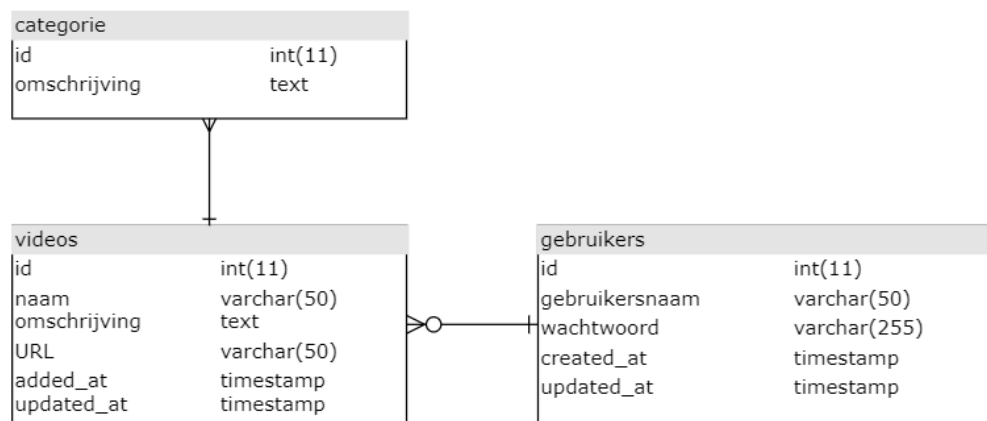
Use case tabel voor de klant inlog functie

<b>Versie nummer</b>	1.0
<b>Use case naam</b>	login
<b>actoren</b>	klant
<b>doel</b>	Klant wilt inloggen op website
<b>triggers</b>	Bestaand klant vult inlog gegevens in
<b>Pre-conditie</b>	Klant heeft inlog gegevens
<b>Post-conditie</b>	Klant is ingelogd
<b>hoofdscenario</b>	1.actor tikt gebruikersnaam in 2.actor stuurt ingevoerde gegevens in 3.systeem check of alles ingevoerde gegevens kloppen 4.systeem slaat gegevens op 5.systeem geeft een bevestiging dat de klant met succes is ingelogd 6.systeem keer terug naar hoofdscherm
<b>Alternatieve scenario's</b>	geen
<b>excepties</b>	1.1 actor annuleert 1.2 systeem keer terug naar hoofdscherm 2.1 actor vult niet alle gegevens in 2.2 systeem geeft hier melding van
<b>kwaliteiten</b>	Bij enige vorm van excepties het systeem laten terug keren in vorige staat

Use case tabel voor de registratie van nieuwe gebruikers.

<b>Versie nummer</b>	1.0
<b>Use case naam</b>	Registreren
<b>actoren</b>	klant
<b>doel</b>	Het aanmaken van een account
<b>triggers</b>	Actor drukt op registreren om een account aan te maken
<b>Pre-conditie</b>	Is op website en alle persoons gegevens zijn ingevoerd
<b>Post-conditie</b>	Klant heeft account aangemaakt
<b>hoofdscenario</b>	1.actor geeft klant gegevens op(gebruikersnaam, wachtwoord) 2. actor drukt op account aanmaken 3.systeem controleert of alle gegevens zijn ingevoerd 4.systeem checkt of email adres van de actor niet al in het systeem staat 5.systeem slaat gegevens op 6.systeem geeft een bevestiging dat registratie gelukt is 7.systeem keert met actor ingelogd terug naar de hoofdscherm
<b>Alternatieve scenario's</b>	geen
<b>excepties</b>	1.1 actor annuleert 2.1actor vult niet alle velden in
<b>kwaliteiten</b>	Bij enige vorm van excepties het systeem laten terug keren in vorige staat

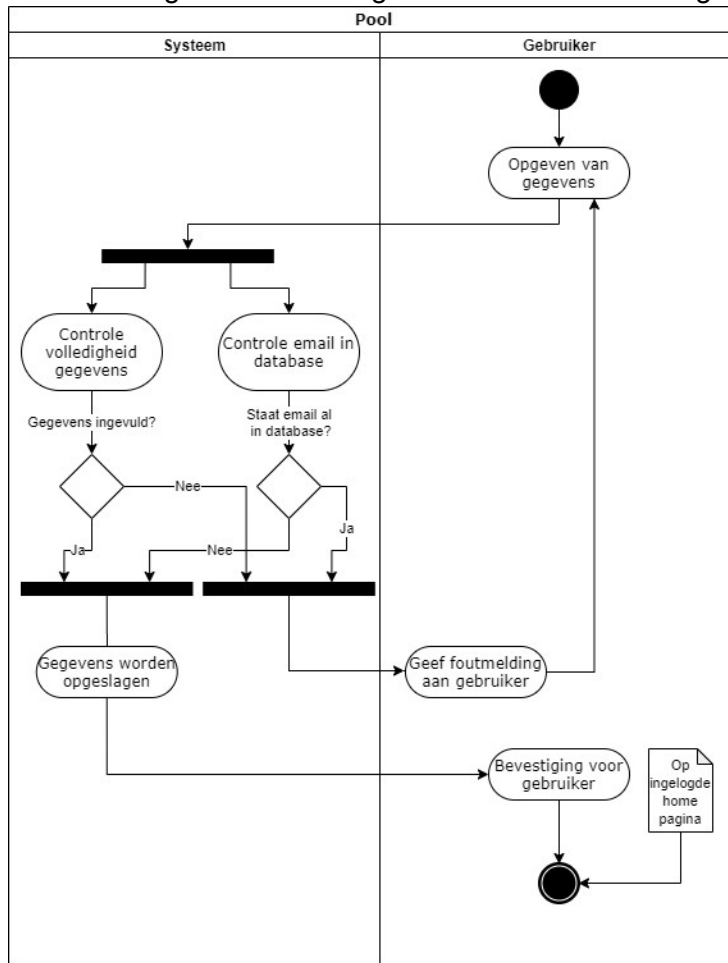
## ERD Diagram





## Activiteit diagrammen en testacties

Activiteit-diagram voor de registratie van een nieuwe gebruiker.



Activiteit-diagram voor het inloggen van een gebruiker.

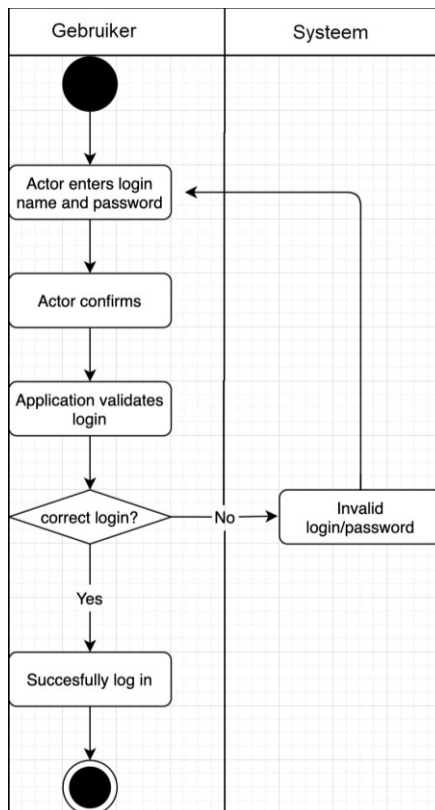
*Testacties:*

**Succesvol registreren:**

1. Opgeven van gegevens
2. Systeem controleert
3. Gegevens zijn opgeslagen
4. Bevestiging gebruiker

**Foutieve registratie:**

1. Opgeven van gegevens
2. Systeem controleert
3. Foutmelding aan gebruiker



### Testacties:

#### Succesvol inloggen:

1. Opgeven van gegevens
2. Systeem controleert
3. Succesvol de gebruiker ingelogd

#### Fout bij inloggen:

1. Opgeven van gegevens
2. Systeem controleert
3. Foute login

```

    usecaseDiagram
        actor Actor
        usecase UC1(( ))
        usecase UC2[Valideer aantal ongeldige login's]
        usecase UC3{Aantal ongeldige login's <= 2}
        usecase UC4[Actor is niet bekend]
        usecase UC5[<<Form>> Inloggen Abonnee]
        usecase UC6[Verhoog aantal ongeldige login's]
        usecase UC7[Actor voert login en password in]
        usecase UC8[Actor Bevestigt]
        usecase UC9[Website Valideert login]
        usecase UC10{Login en password geldig}
        usecase UC11[Markeer actor als abonnee]
        usecase UC12[Actor meld zich af]
        usecase UC13[Actor meld zich af]
        usecase UC14[Acteur drukt op filter knop]
        usecase UC15[Acteur typt de video course in die hij/zij wilt zien]
        usecase UC16[Acteur drukt op enter]
        usecase UC17[Systeem slaat ingevoerde gegevens in]
        usecase UC18[Acteur meld zich af]
        usecase UC19[Acteur meld zich af]
        usecase UC20[Systeem laat ingetypte Video aan de actor zien]
        usecase UC21[Acteur sluit website]

        UC1 --> UC2
        UC2 --> UC3
        UC3 --> UC4
        UC3 --> UC5
        UC5 --> UC7
        UC7 --> UC8
        UC8 --> UC9
        UC9 --> UC10
        UC10 --> UC6
        UC6 --> UC2
        UC10 --> UC11
        UC11 --> UC14
        UC12 --> UC1
        UC13 --> UC1
        UC14 --> UC15
        UC15 --> UC16
        UC16 --> UC17
        UC17 --> UC18
        UC18 --> UC1
        UC17 --> UC19
        UC19 --> UC20
        UC20 --> UC21
        UC21 --> UC1
  
```

The diagram illustrates the login and video filtering process for a user. It is divided into two scenarios: the Main Scenario (Blauw) and the Alternative Scenario (Rood).

**Main Scenario (Blauw):**

- The process starts with a start node (white circle).
- Use Case: Valideer aantal ongeldige login's (white rounded rectangle).
- Decision: Aantal ongeldige login's <= 2 (white diamond).
- If the condition is true, the flow goes to: Actor is niet bekend (white rounded rectangle).
- If the condition is false, the flow goes to: <<Form>> Inloggen Abonnee (white rounded rectangle).
- Use Case: Actor voert login en password in (white rounded rectangle).
- Use Case: Actor Bevestigt (white rounded rectangle).
- Use Case: Website Valideert login (white rounded rectangle).
- Decision: Login en password geldig (white diamond).
- If the condition is true, the flow goes to: Markeer actor als abonnee (white rounded rectangle).
- If the condition is false, the flow goes to: Verhoog aantal ongeldige login's (white rounded rectangle), which loops back to the start of the validation process.
- The process ends at a final node (white circle).

**Alternative Scenario (Rood):**

- Use Case: Actor meld zich af (red rounded rectangle).
- Use Case: Actor meld zich af (red rounded rectangle).
- Use Case: Acteur drukt op filter knop (white rounded rectangle).
- Use Case: Acteur typt de video course in die hij/zij wilt zien (white rounded rectangle).
- Use Case: Acteur drukt op enter (white rounded rectangle).
- Use Case: Systeem slaat ingevoerde gegevens in (white rounded rectangle).
- Use Case: Actor meld zich af (red rounded rectangle).
- Use Case: Systeem laat ingetypte Video aan de actor zien (white rounded rectangle).
- Use Case: Actor sluit website (red rounded rectangle).

**Flow Details:**

- The flow starts with the Actor logging out (Rood) and returning to the start node.
- The Actor presses the filter button (Blauw), which leads to typing the video course name.
- The Actor presses enter, and the system saves the input.
- The Actor logs out again (Rood).
- The system displays the typed video to the Actor.
- The Actor closes the website (Rood), returning to the start node.

### Succesvol zoeken:

- Fout bij toegang zoekfunctie, gebruiker is niet bekend:**

- 11

**Succesvolle toegang zoekfunctie, met enkel login fout van de gebruiker:**

1. Inloggen gebruiker
2. Systeem controleert/valideert
3. Foute inloggegevens ingegeven door gebruiker
4. Inloggen gebruiker
5. Systeem controleert/valideert
6. Input gebruiker om video's te filteren
7. Systeem slaat gegevens op en laat zoekresultaat zien aan gebruiker
8. Gebruiker heeft toegang tot de juiste video