|  |
| --- |
| Einddocument  WPL-project |
| **Departement: Media, Design & IT**  **Opleiding: Graduaat Programmeren**  **Academiejaar: 2025-2026**  **Door *Vermeld hier alle groepsleden*** |

**Inhoud**

[Einddocument 0](#_Toc215658826)

[WPL-project 0](#_Toc215658827)

[1 Projectbeschrijving 2](#_Toc215658828)

[1.1 Projectinhoud 2](#_Toc215658829)

[1.2 Groepsnaam 2](#_Toc215658830)

[1.3 Gebruikershandleiding 2](#_Toc215658831)

[1.3.1 Richtlijnen 2](#_Toc215658832)

[2 Projectrealisatie 3](#_Toc215658833)

[3 Technische Documentatie 3](#_Toc215658834)

[3.1 Front-end 3](#_Toc215658835)

[3.1.1 Architectuur 3](#_Toc215658836)

[3.1.2 Technologieën & Tools 3](#_Toc215658837)

[3.1.3 Styling en UX 3](#_Toc215658838)

[3.2 Backend 3](#_Toc215658839)

[3.2.1 Architectuur 3](#_Toc215658840)

[3.2.2 Technologieën en Tools 3](#_Toc215658841)

[3.2.3 Database 4](#_Toc215658842)

[3.2.4 Versiebeheer 4](#_Toc215658843)

[3.2.5 Deployment en Hosting 4](#_Toc215658844)

[3.2.6 Security 4](#_Toc215658845)

[4 Projectmethodologie 4](#_Toc215658846)

[4.1 Scrum 4](#_Toc215658847)

[4.2 Scrum-events 4](#_Toc215658848)

[4.3 Tijdlijn 4](#_Toc215658849)

[4.4 Use Cases 4](#_Toc215658850)

[4.5 Tools en samenwerking 4](#_Toc215658851)

[5 Reflecties 5](#_Toc215658852)

[6 Bijlages 5](#_Toc215658853)

# Projectbeschrijving

## Projectinhoud

*Beschrijf hier kort wat je hebt gebouwd (voeg opgave toe of verkorte omschrijving van de opgave).*

## Groepsnaam

*Vermeld hier de groepsnaam*

## Gebruikershandleiding

*In dit onderdeel geef je een korte beschrijving van hoe een gebruiker de applicatie kan gebruiken. Richt je op de belangrijkste functionaliteiten en scenario’s, zonder een volledige gedetailleerde handleiding uit te werken.*

### Richtlijnen

1. *Voeg de belangrijkste scenario’s (flows) toe  
   Beschrijf de kernscenario’s die een gebruiker doorloopt in de applicatie, bijvoorbeeld:*

* *Inloggen / registreren*
* *Navigeren door het startscherm*
* *…*

1. *Resultaat van de acties  
   Geef aan wat er gebeurt wanneer een gebruiker een actie uitvoert, bijvoorbeeld: gegevens worden opgeslagen, een taak wordt afgerond, …*
2. *Visualiseer je uitleg  
   Voeg voor elk scenario een screenshot of afbeelding toe zodat de handleiding visueel duidelijk is.*

*Het doel is dat iemand die je handleiding leest een goed overzicht krijgt van hoe de applicatie werkt, welke stappen een gebruiker kan volgen en wat het resultaat van zijn acties is, zonder dat alle details tot in de kleinste knop worden beschreven.*

# Projectrealisatie

*Vermeld hier:*

1. *De url naar jouw gedeployde applicatie:*
2. *De link naar de github repository met de definitieve versie:*
3. *De link naar de tool voor projectbeheer (Jira) met de volledige projectbacklog:*

# Technische Documentatie

## Front-end

### Architectuur

*Beschrijf hoe de bestandsstructuur van de frontend is opgebouwd. Leg uit hoe de mappen en bestanden zijn georganiseerd.*

*Geef daarnaast aan hoe deze structuur bijdraagt aan overzicht, onderhoudbaarheid en schaalbaarheid van de frontend.*

### Technologieën & Tools

*In dit onderdeel beschrijf je welke technologieën en tools zijn gebruikt voor de ontwikkeling van het project, en indien relevant, waarom deze keuzes zijn gemaakt. Dit kan bijvoorbeeld betrekking hebben op programmeertalen, frameworks, templating engines en package managers.*

*Indien gewenst kan hier ook kort worden vermeld waarom bepaalde alternatieven niet zijn gekozen, bijvoorbeeld waarom EJS is gekozen in plaats van een ander templating framework.*

Voor de ontwikkeling van dit project (een pizzashop webapplicatie) hebben we gekozen voor een **full-stack JavaScript/TypeScript stack** met Node.js. Dit biedt gangbare taalgebruik aan zowel front- als backend, wat de ontwikkeling versnelt en fouten vermindert in een teamproject.

**Belangrijkste technologieën en tools:**

* **Programmeertaal**: TypeScript (broncode in src/) → naar JavaScript.

Keuze voor TypeScript omdat het type-safety biedt, betere autocomplete in IDE en schaalbaarder is dan plain JavaScript.

* **Backend framework**: Express.js → voor routing, middleware en server-logica. Simpel, lichtgewicht en perfect voor een schoolproject.
* **Templating**: EJS (templates in views/) → voor dynamische HTML-generatie (bijv. productlijsten uit database). **Waarom EJS?** Het is minimalistisch, integreert naadloos met Express, en voelt dicht bij pure HTML aan dus ideaal voor beginners.
* **Package manager**: npm → met package.json voor dependencies (Express, ejs, dotenv) en scripts (npm run dev, npm run build).
* **Development tools**:
  + Nodemon → voor live-reload tijdens ontwikkeling.
  + ts-node → om TypeScript direct te draaien zonder build-stap.
* **Static assets**: Geserveerd vanuit public/ (css, js, images).

### Styling en UX

*Hoe is de styling gebeurd? (bv. css flexbox, css grid,…)*

*Is de styling responsief ?  
Is er een designsysteem gebruikt of is de styling custom ?*

De styling is volledig custom gebouwd met pure CSS, zonder externe UI-frameworks zoals Bootstrap of Tailwind. Dit geeft maximale controle over het unieke pizzathema.

**Hoe is de styling gebeurd?**

* **CSS Grid** → voor de globale pagina-layout (header, main, footer, flash messages).
* **Flexbox** → voor componenten zoals navigatie, productkaarten en winkelmandje-tabellen.
* **CSS Custom Properties** (variabelen in :root) → centrale plek voor kleuren, spacing, border-radius en fonts → makkelijk aan te passen en consistent over de hele site.
* Micro-interacties → zoals hover-effects (scale & kleurtransitie) op knoppen voor een aangenaam gevoel.

**Is de styling responsief?** Ja, volledig mobiel-first met media queries:

* Breakpoints op 1024px (desktop), 768px (tablet) en <767px (smartphone).
* Hamburger-menu verschijnt onder 1000px voor mobiele navigatie.
* Grids en flex-items passen zich automatisch aan.

**Designsysteem?** Geen extern designsysteem (zoals Material Design), maar een **eigen custom designsysteem** opgebouwd met CSS-variabelen en modulaire CSS-files in public/css/. Dit zorgt voor consistentie (één plek voor primaire kleur #c40009) en eenvoudige uitbreiding.

Resultaat: een snelle, aantrekkelijke en gebruiksvriendelijke UX die eetlust opwekt stimuleert, perfect afgestemd op een pizza-bestelwebsite.

## Backend

### Architectuur

*Beschrijf hoe de bestandsstructuur van de frontend is opgebouwd. Leg uit hoe de mappen en bestanden zijn georganiseerd.*

*Geef daarnaast aan hoe deze structuur bijdraagt aan overzicht, onderhoudbaarheid en schaalbaarheid van de backend.*

### Technologieën en Tools

*In deze sectie beschrijf je welke technologieën en tools voor het project zijn gekozen, waarom deze geschikt zijn en hoe ze bijdragen aan de stabiliteit, schaalbaarheid en onderhoudbaarheid van het project.*

*Daarbij kan ook worden toegelicht of er alternatieven zijn overwogen, bijvoorbeeld MySQL versus MongoDB, en waarom uiteindelijk voor één van deze opties is gekozen.*

### Database

*Bespreek hier in detail welke database je hebt gebruikt en waarom. Voeg hierbij het databaschema toe (met relaties in bijlage en verwijs hiernaar).*

### Versiebeheer

*In dit onderdeel leg je uit hoe versiebeheer is toegepast binnen het project. Bijvoorbeeld:*

* *Welke tools zijn gebruikt (bijv. Git, GitHub).*
* *Welke workflow is gevolgd (bijv. feature branches, pull requests, main branch).*
* *Hoe werd er omgegaan met merge conflicts*
* *Hoe versiebeheer bijdraagt aan samenwerking, codekwaliteit en traceerbaarheid van wijzigingen.*

### Deployment en Hosting

*Leg hier uit waar je backend is gedeployd. Beschrijf de volgende punten:*

*1. Hostingomgeving (bv. Vercel)*

*2. Deploymentproces: Deployt de code automatisch of is dit een handmatig proces?*

*3. Omgevingsvariabelen: Waar worden omgevingsvariabelen bijgehouden?*

### Security

*In dit onderdeel beschrijf je welke beveiligingsmethoden je hebt gebruikt. Voeg hierbij concrete voorbeelden toe uit je project. Bv. JWT (authenticatie), gebruik van rollen (autorisatie), encryptie (https), input validatie / sanitatie,…*

# Projectmethodologie

## Scrum

*Beschrijf kort wat scrum is en welke voordelen deze methodologie heeft voor je project.*

## Scrum-events

*Licht kort toe welke scrum-events jullie hebben gebruikt en beschrijf deze. Voeg 2 voorbeelden toe van de sprint retrospective in bijlage.*

## Tijdlijn

*Geef een visueel overzicht van de sprints en epics. Je mag deze toevoegen in de bijlage (printscreens) of hier vermelden.*

## Use Cases

*Welke use cases heb je gerealiseerd?*

*Hoe heb je de prioriteiten bepaald van de usecase?*

*Welke usecases had je nog graag geïmplementeerd?*

## Tools en samenwerking

*Vermeld welke tools er gebruikt zijn voor taakbeheer en samenwerking (bv. Jira)*

# Reflecties

*Voeg hier per teamlid een reflectiepagina toe.*

*Elk teamlid reflecteert over het verloop van de opdracht en zijn/haar bijdrage tot het project:*

* + *Welk was voor jou individueel de grootste challenge? Hoe ben je daarmee omgegaan?*
  + *Noem 2 leerpunten die je hebt bijgeleerd.*
  + *Noem 2 werkpunten die je voor jezelf hebt gedefinieerd op basis van het verloop van het project, waar je in de toekomst belang aan hecht om zeker nog bij te leren.*
  + *Hoe ervaarde je het werken in teamverband aan dit project? Zou je voorkeur ernaar uitgaan om het project alleen te realiseren, mocht je dit traject kunnen overdoen? Waarom wel / niet?*

# Bijlages

*Voeg hier jullie bijlages toe waar en verwijs hiernaar.*

* *1. Wat was voor jou individueel de grootste Challenge? Hoe ben je daarmee omgegaan?*

*Antwoord:*

*Voor mij was het ritme behouden om mee te zijn met het project wel een Uitdaging,*

*ben twee weken afwezig geweest wegens persoonlijke reden en dat maakte dat ik niet mee was met de sprint’s waardoor ik achterliep met de progress dat was gemaakt. Dit creëerde ook een probleem dat ik de website niet meer kon runnen.*

*Hoe ik er mee om ga:*

*Het heeft me verteld dat ik zelfs op moeilijke momenten moet volgen als je in een team samen werkt aan een project en je bestanden moet up to date houden.*

* *2. Noem 2 leerpunten die je hebt bijgeleerd.\*

*Antwoord:*

*Leerpunt 1:*

*Duidelijke communicatie tussen collega’s over bepaalde taken, brede uitleg wat er moet gebeuren, prioriteiten stellen tussen belangrijke onderdelen die eerst moeten gedaan worden voor een volgende stappen ondernomen worden.*

*Leerpunt 2:*

*Dat een volledige website bouwen groter is dan ik verwachte, er komt meer bij kijken dan ik verwachte alle verschillende onderdelen die moeten samen werken om de webiste te laten functioneren zoals behoren.*

* *3. Noem 2 werkpunten die je voor jezelf hebt gedefinieerd.*

*Antwoord:*

*Werkpunt 1:*

*Me meer focussen op een goede basis waar we kunnen van vertrekken als ik nog eens een website moet bouwen in team verband, hier bedoel ik mee dat ik me tijd me zeker meer zal verdiepen in hoe je het beste kunt starten met bepaalde zaken bij het opstarten van het bouwen aan een website*

*Werkpunt 2:*

Het beter kunnen uitleggen, het verduidelijken van mijn start,progress,end van een onderdeel dat te maken heeft met de website (project).

* *4. Hoe ervaarde je het werken in teamverband aan dit project? ? Zou je voorkeur ernaar uitgaan om het project alleen te realiseren, mocht je dit traject kunnen overdoen?*

*Antwoord:*

*Waarom wel / niet?:*

*Niet:*

*Mijn voorkeur zou niet direct overgaan naar het zelf maken van een website wegens dat ik nog niet zo goed weet wat een website echt allemaal nodig heeft.*

*Het is fijn om met mensen samen te kunnen werken wat daar leer je ook veel uit van andere hun ideeën en manier van werken.*

Wel:

Alleen kan je alles laten werken hoe jij het wilt als je alles ook begrijpt wat nodig is om een website te kunnen maken. Het heeft zeker voordelen om het alleen te doen maar het kom er op neer hoe goed je zelf bent dat dit een voordeel is of niet.