2025

in(Forum)

FORUM DETAILS

CIARA HENRIQUEZ

Inhoudspagina

Inhoud

Inhoudspagina 0
1.1 Algemene informatie 2
1.2 Huidige situatie 2
1.3 Gewenste situatie 3
1.4 Eisen en wensen/Behoefte analyse 4
2. Functioneel Ontwerp 5
2.1 Doel van het systeem 5
2.2 Functionele specificaties 6
2.3 Gebruikersinteractie en UX 8
2.4 Data en processen 11
2.5 Integraties en koppelingen 12
2.6 Globale planning: 12
2.7 Contact 13
3. Technisch Ontwerp 14
3.1 Systeemarchitectuur 14
3.2 Databaseontwerp 15
3.3 Beveiliging en authenticatie 16
Prestatie- en beschikbaarheidseisen 22
3.5 Deployment en onderhoud 22
4 Testplan 23
5. Verbetervoorstellen 25
6 D.O.D 26
Versiebeheer 27
Bronnen 29
Bijlagen 29

1.1 Algemene informatie

- Wat is de naam van het project?

- In(forum)

- Wie zijn de belanghebbenden?

- Stagiaire: Ciara Henriquez
- Begeleider: Hugo Delsing
- Open doelgroep en niet specifiek gericht

- Wat is de aanleiding van het project

- Gebruikers moeten accounts kunnen aanmaken, inloggen, berichten plaatsen en reacties toevoegen dit is nodig op een bepaald platform namelijk: in een forumomgeving.

- Wat is het doel van het project?

- Dieper inzicht krijgen in databasebeheer en web functionaliteiten. Dit project dient als een leerervaring om praktijkervaring op te doen met PHP, MySQL en het bouwen van interactieve webapplicaties. Het systeem moet gebruikers in staat stellen veilig en gemakkelijk te communiceren.

1.2 Huidige situatie

- Hoe wordt het probleem momenteel opgelost

- Ik heb onderzoek gedaan naar hoe gebruikers met elkaar kunnen communiceren op een efficiënte manier. Bij bestaande oplossingen ontbreken functies zoals gebruiksvriendelijke profielbeheeropties, een intuïtief reactie- en berichtensysteem en een robuust beveiligingsmechanisme

- Welke knelpunten ervaren gebruikers met de huidige werkwijze

- Er is niet genoeg diversiteit in opties. Gebruikers ervaren dat profielbeheer niet intuïtief is, reactiesysteem is beperkt en beveiliging onvoldoende

- Zijn er bestaande systemen of processen die hiermee verband houden

- FOK!forum
- Tweakers

1.3 Gewenste situatie

- Welke verbeteringen moeten worden gerealiseerd?

- Het moet makkelijk te navigeren zijn
- Het moet goed beveiligd zijn
- Het is voor iedereen toegankelijk

- Wat zijn de functionele behoeften van gebruikers?

- **Registratie en login**
  Ik ontwikkelde een veilige gebruikersregistratie en loginfunctionaliteit met PHP en MySQL. Dit omvatte onder andere het versleutelen van wachtwoorden om gegevens te beschermen.
- **Gebruikersprofielen**
  Functionaliteiten gebouwd waarmee gebruikers hun profielgegevens, zoals avatar en biografie, konden aanpassen. Deze gegevens werden opgeslagen en up-to-date gehouden in de database.
- **Reactiesysteem**
  Ik implementeerde een systeem waarmee gebruikers reacties konden plaatsen, bewerken en verwijderen. Dit vergde een robuuste database-structuur om gegevens consistent en efficiënt te beheren.

- Wat zijn de technische behoeften (bijv. integraties, compatibiliteit)

- Databaseondersteuning (MySQL)
- Ondersteuning voor PHP als back-end taal
- Mogelijkheid tot uitbreiding met API's in de toekomst

- Zijn er wettelijke of beleidsmatige vereisten waaraan moet worden voldaan

- “AVG-compliance voor het opslaan van persoonsgegevens”. Dit betekent dat gebruikers expliciet akkoord moeten gaan met gegevensopslag en er maatregelen moeten worden genomen om persoonsgegevens te beschermen.

1.4 Eisen en wensen/Behoefte analyse

- Wat zijn de harde eisen waaraan het systeem moet voldoen? (MoSCoW)

Must have	Should have	Could have	Won't have
Kunnen inloggen	Kunnen liken	Post count	Chatfunctie
Kunnen plaatsen	Kunnen reageren	Thema's aanpassen	Live notificaties
Profiel bewerken	Categorieën bekijken	Eigen avatar maken	Video-ondersteuning
		Instellingen	
		Andere profielweergave	

- Welke wensen zijn er die als optioneel kunnen worden beschouwd

- Een eigen avatar maken
- Meldingen
- Chatfunctie

- Wat zijn de prestatie-eisen (bijv. snelheid, betrouwbaarheid)

- Betrouwbaar
- Duidelijk structuur
- Toegang

- Welke beveiligingseisen gelden er voor het systeem

- Gegevens veilig opslaan
- Bescherming tegen **SQL-injecties en XSS**
- Versleuteling van wachtwoorden met **bycrpt**

2. Functioneel Ontwerp

2.1 Doel van het systeem

- Wat is de primaire functionaliteit van het systeem

- **Registratie en login**
- **Gebruikersprofielen**
- **Reactiesysteem**

- Welke gebruikersgroepen maken gebruik van het systeem

- Open doelgroep (iedereen met interesse in de onderwerpen)
- Studenten
- Fanclubs
- Werknemers

- Wat zijn de belangrijkste user cases?

Registratie:

  - Als nieuwe gebruiker wil ik een account kunnen aanmaken zodat ik berichten kan plaatsen en reageren. Het registratieproces omvat het invullen van een formulier met velden zoals naam, e-mailadres, en wachtwoord. Wachtwoorden worden versleuteld opgeslagen in de database om de veiligheid van gebruikersgegevens te waarborgen.

Acceptatiecriteria:

- Het registratieformulier bevat velden voor naam, e-mail en wachtwoord.
- Het wachtwoord wordt versleuteld opgeslagen met bcrypt.
- Na registratie ontvangt de gebruiker een bevestigingsmail (optioneel).
- Foutmeldingen verschijnen bij ontbrekende of onjuiste invoer.
- Prioriteit: *Must have*

Inloggen:

  - Als geregistreerde gebruiker wil ik kunnen inloggen, zodat ik toegang krijg tot mijn profiel en kan deelnemen aan discussies. De inlogpagina bevat velden voor gebruikersnaam en wachtwoord, met validatie van gegevens en versleuteling van wachtwoorden.

Acceptatiecriteria:

- Inlogformulier valideert invoer en geeft duidelijke foutmeldingen.
- Wachtwoord wordt versleuteld vergeleken met database.
- Prioriteit: *Must have*

Profiel bewerken:

- Als gebruiker wil ik mijn profielgegevens kunnen aanpassen, zodat anderen meer over mij te weten komen. Gebruikers kunnen hun avatar en biografie aanpassen via een profielpagina, waarbij de nieuwe gegevens naar de database worden gestuurd en geüpdatet.

Profielfoto instellen:

- Als gebruiker wil ik een profielfoto kunnen uploaden, zodat mijn profiel een persoonlijke uitstraling krijgt. Gebruikers kunnen een foto kiezen van hun apparaat, die vervolgens wordt geüpload naar de server en gekoppeld aan hun account.

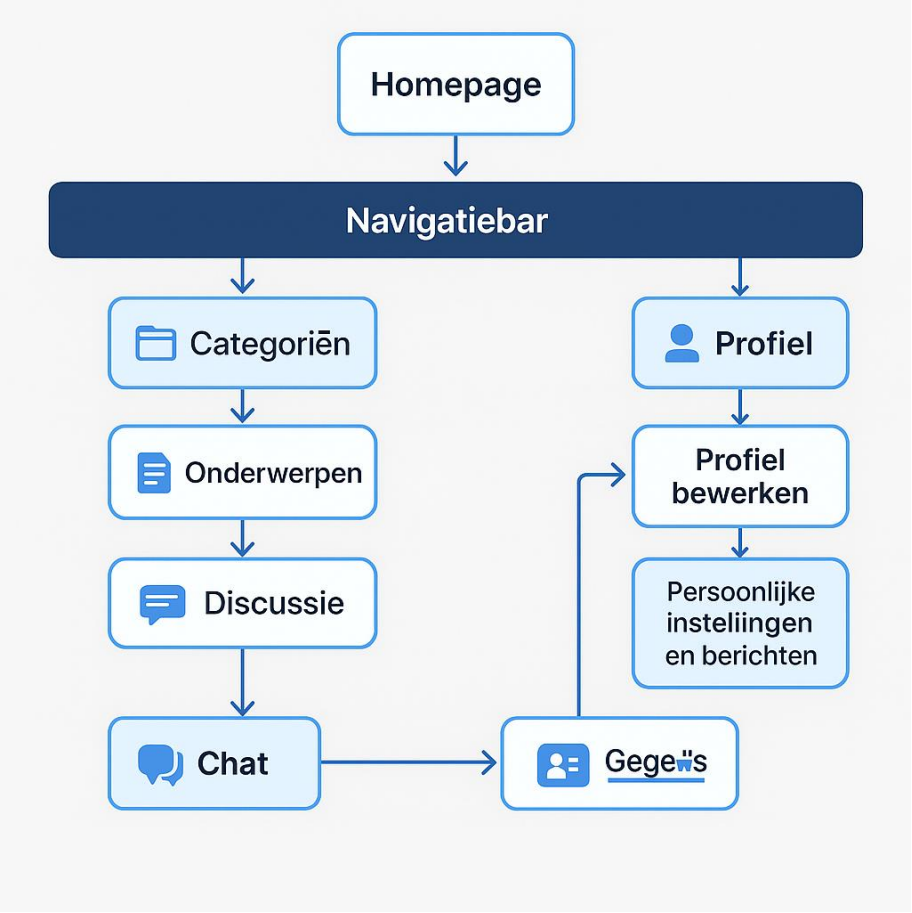
Acceptatiecriteria:

- Upload via bestandselectie van jpg/png.
- Foto wordt opgeslagen op de server en gekoppeld aan account.
- Foutmelding bij te grote bestanden of verkeerd bestandstype.
- Prioriteit: *Should have*

2.2 Functionele specificaties

- Welke hoofdfunctionaliteiten moet het systeem bevatten
- **Categorieën bekijken:**
  Als gebruiker wil ik forumcategorieën kunnen bekijken, zodat ik snel onderwerpen kan vinden die mij interesseren.
- **Nieuw onderwerp aanmaken:**
  Als gebruiker wil ik een nieuw onderwerp kunnen starten, zodat ik een discussie kan beginnen over een bepaald onderwerp.
- **Reageren op berichten:**
  Als gebruiker wil ik kunnen reageren op bestaande berichten, zodat ik kan deelnemen aan gesprekken.
- **Berichten bewerken/verwijderen:**
  Als gebruiker wil ik mijn eigen berichten kunnen bewerken of verwijderen, zodat ik fouten kan corrigeren of ongewenste inhoud kan verwijderen

- Welke schermen en interfaces zijn vereist & Hoe moet de gebruiker door het systeem navigeren



1. Homepage → Overzicht van de categorieën
2. Categorieën → Onderwerpen binnen een bepaalde categorie
3. Onderwerpen → Discussies tussen gebruikers
4. Chat
5. Reactie

-

1. Navigatiebar met opties en onderwerpen
2. Één keus is een artikel per gekozen onderwerp
3. Optie voor profiel
4. Profiel kan worden bewerkt → Persoonlijke instellingen en berichten
5. Opgeslagen profiel is zichtbaar voor anderen

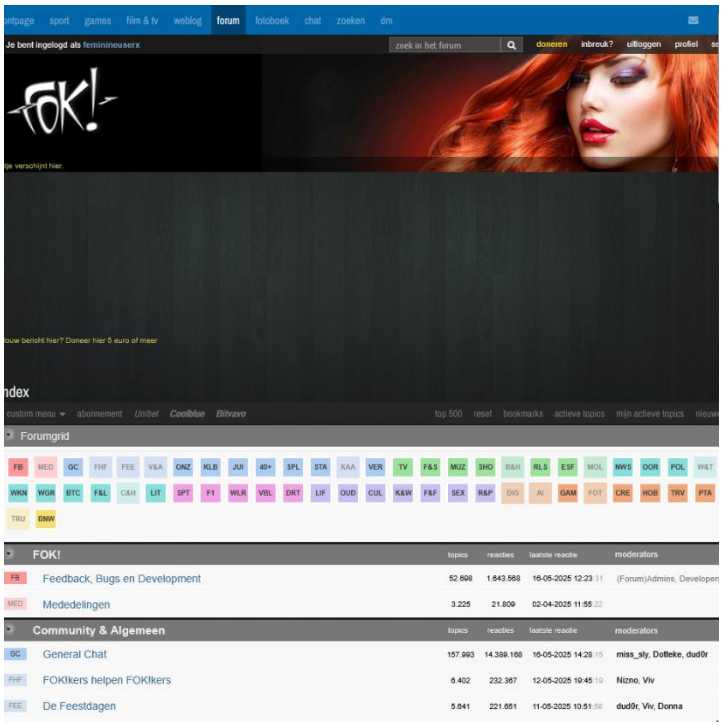
- Welke invoer en uitvoer moet het systeem ondersteunen
- Gebruikersinvoer (berichten, reacties, profielgegevens)

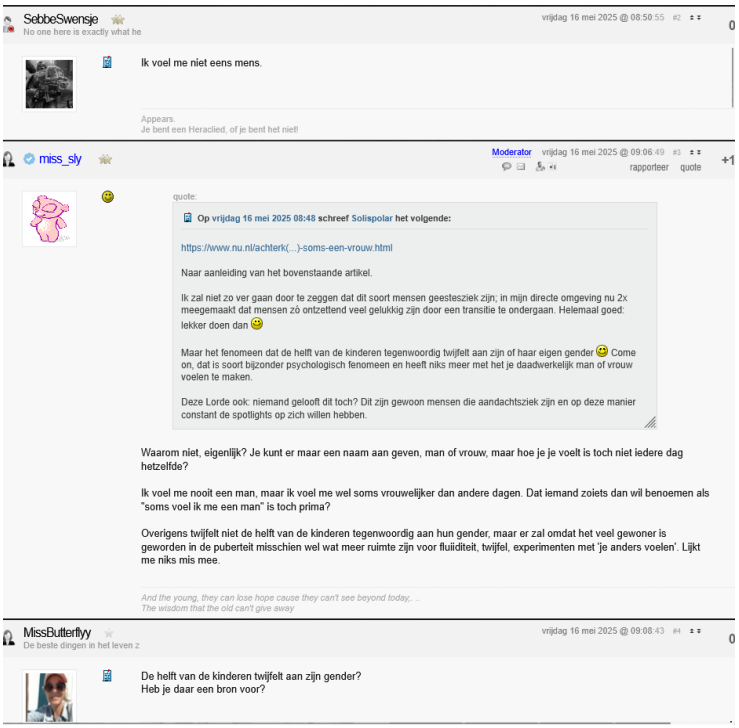
2.3 Gebruikersinteractie en UX

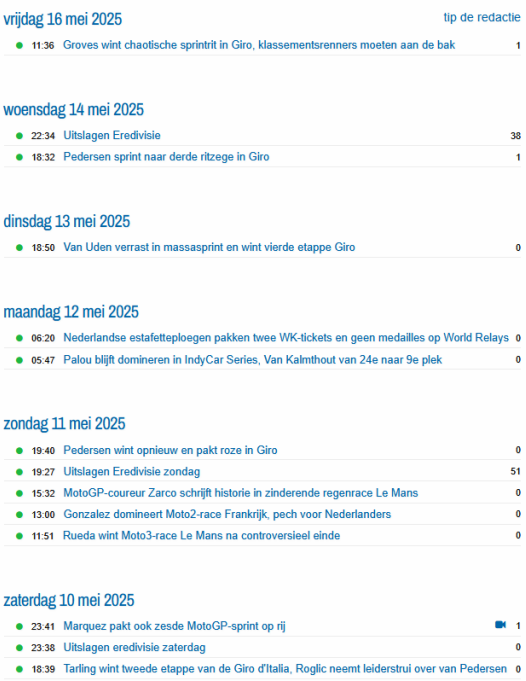
- Hoe moet de gebruikerservaring eruitzien
- Het moet een versie en functionaliteiten hebben als de volgende voorbeelden:

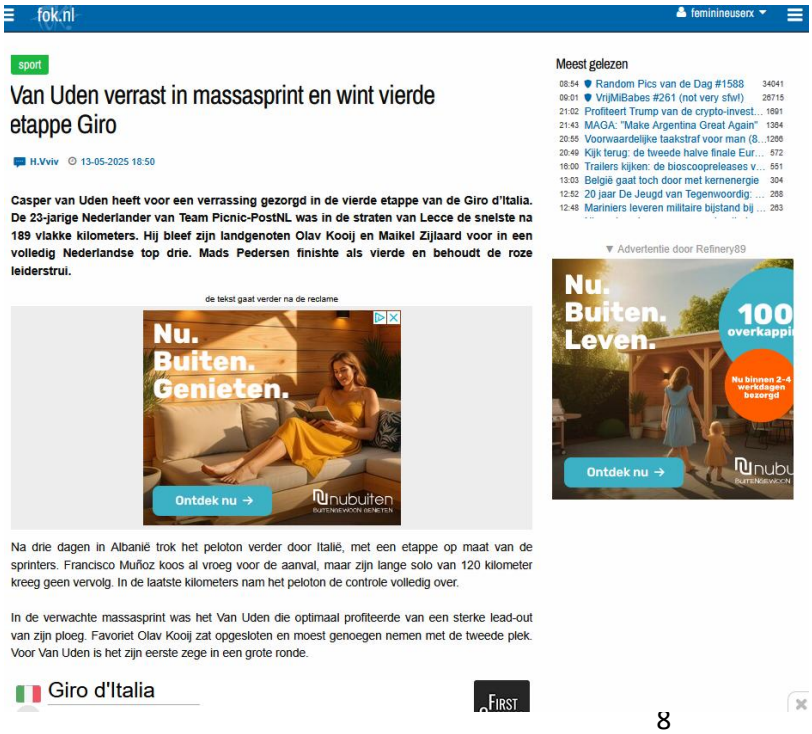
  - https://forum.fok.nl/
  - https://gathering.tweakers.net/

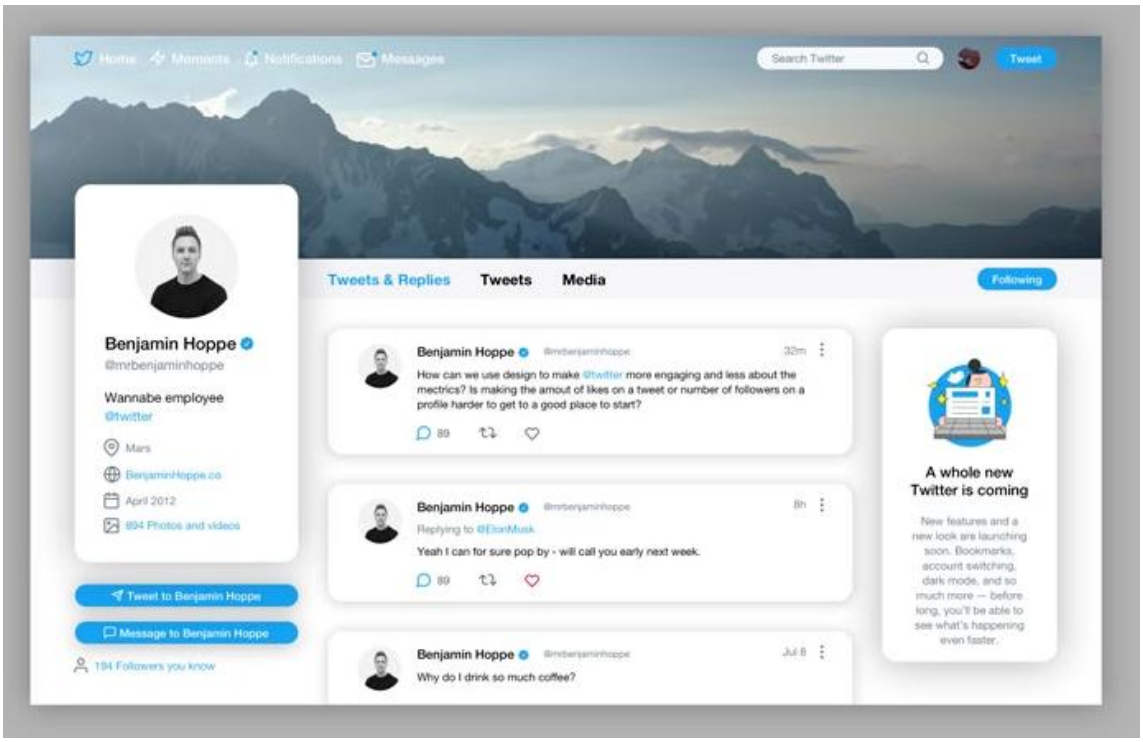
Ik heb deze als voorbeeld genomen:

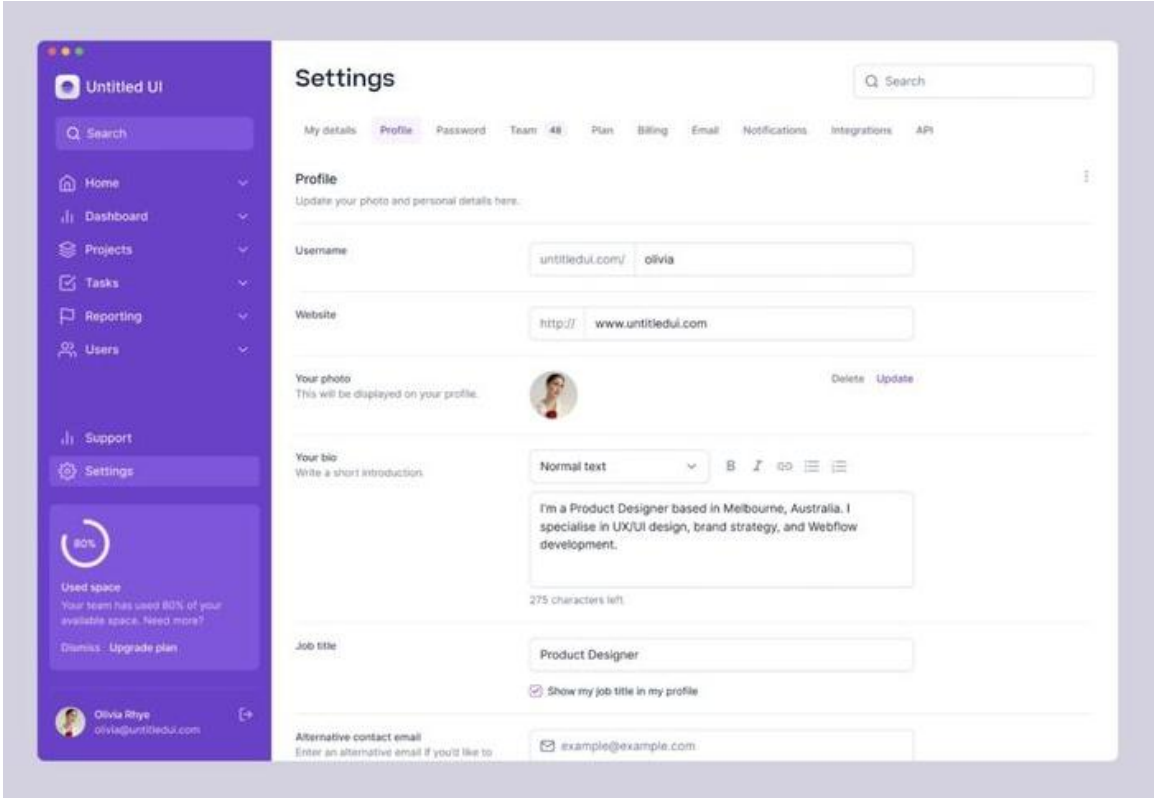


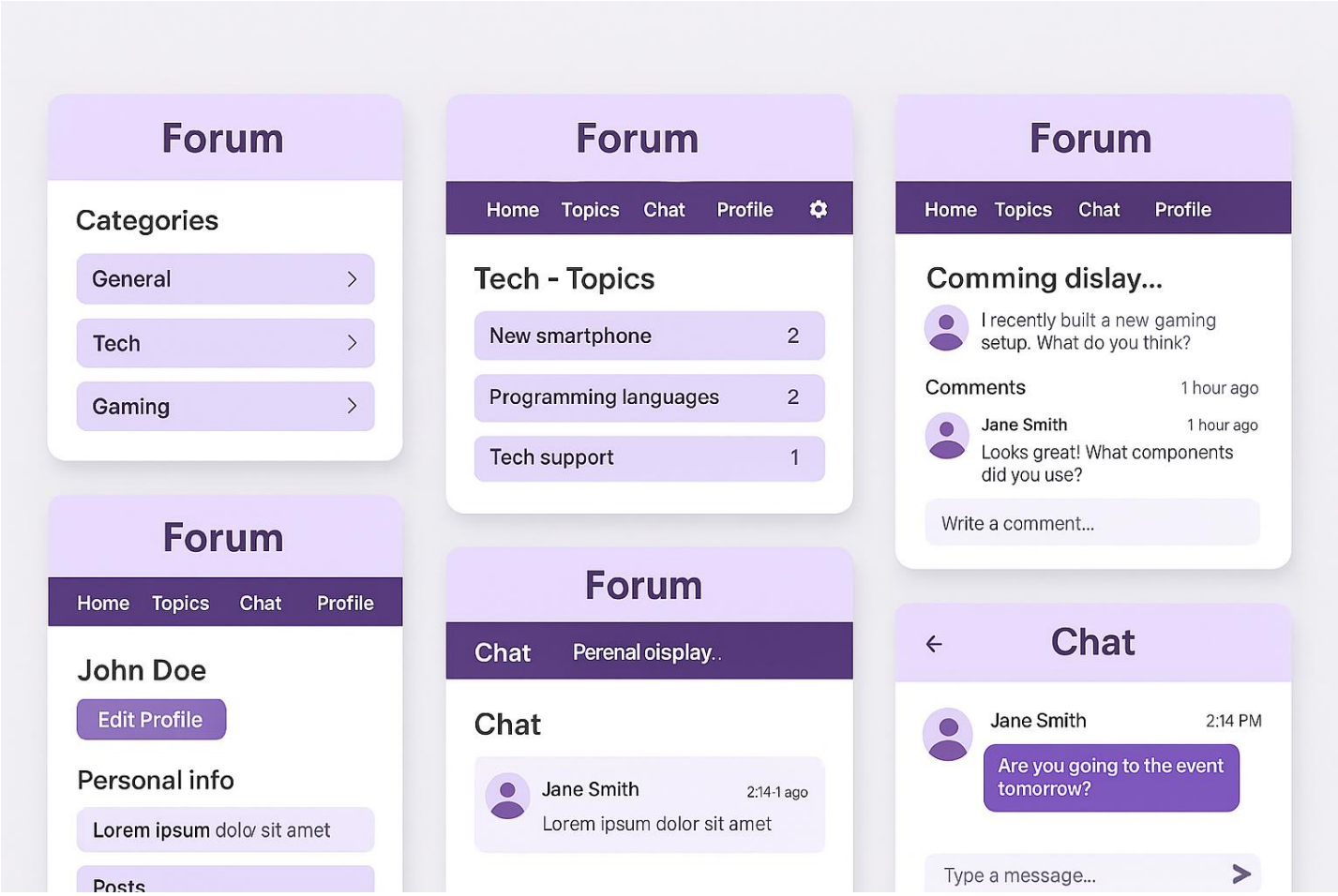







Home
Moments
Notifications
Messages
Search Twitter
Tweet
Benjamin Hoppe
@mrbenjaminhoppe
Wannabe employee
@twitter
Mars
BenjaminHoppe.co
April 2012
894 Photos and videos
Tweet to Benjamin Hoppe
Message to Benjamin Hoppe
194 Followers you know
Tweets & Replies
Tweets
Media
Following
Benjamin Hoppe
@mrbenjaminhoppe
32m
How can we use design to make @twitter more engaging and less about the mectrics? Is making the amout of likes on a tweet or number of followers on a profile harder to get to a good place to start?
89
Benjamin Hoppe
@mrbenjaminhoppe
8h
Replying to @ElonMusk
Yeah I can for sure pop by - will call you early next week.
89
Benjamin Hoppe
@mrbenjaminhoppe
Jul 8
Why do I drink so much coffee?
A whole new Twitter is coming
New features and a new look are launching soon. Bookmarks, account switching, dark mode, and so much more — before long, you'll be able to see what's happening even faster.


Untitled UI
Search
Home
Dashboard
Projects
Tasks
Reporting
Users
Support
Settings
80%
Used space
Your team has used 80% of your available space. Need more?
Dismiss
Upgrade plan
Olivia Rhye
olivia@untitledui.com
Settings
Search
My details
Profile
Password
Team 48
Plan
Billing
Email
Notifications
Integrations
API
Profile
Update your photo and personal details here.
Username
untitledui.com/
olivia
Website
http://
www.untitledui.com
Your photo
This will be displayed on your profile.
Delete
Update
Your bio
Write a short introduction.
Normal text
I'm a Product Designer based in Melbourne, Australia. I specialise in UX/UI design, brand strategy, and Webflow development.
275 characters left
Job title
Product Designer
Show my job title in my profile
Alternative contact email
Enter an alternative email if you'd like to
example@example.com

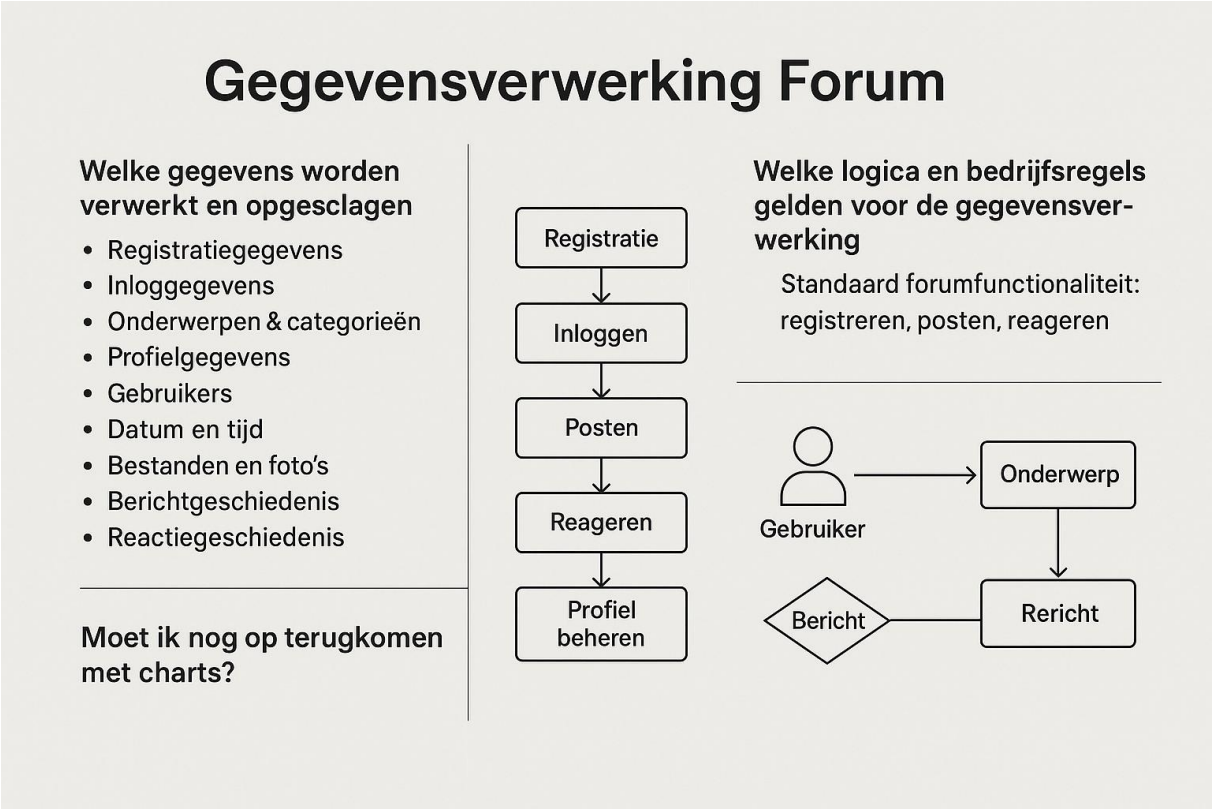


Het gegokte en verwachte resultaat in gemaakte schets binnen mijn project:

- Zijn er specifieke UI/UX richtlijnen of beperkingen
    - Geen specifieke richtlijnen
- Welke toegankelijkheidseisen gelden er
    - Basis toegankelijkheid voor schermlezers en toetsenbordnavigatie

2.4 Data en processen

- Welke gegevens worden verwerkt en opgeslagen
    - Registratiegegevens
    - Inlog gegevens
    - Onderwerpen & Categorieën
    - Profiel gegevens
    - Gebruikers
    - Datum en tijd
    - Bestanden en foto's
    - Bericht geschiedenis
    - Reactie geschiedenis
- Welke workflow en processen moeten worden ondersteund



2.5 Integraties en koppelingen

- Moet het systeem integreren met andere software of databronnen?
    - Niet vereist op dit moment
- Welke API's of protocollen worden gebruikt voor integratie?
    - Momenteel geen

2.6 Globale planning:

Deze planning biedt een gedetailleerd overzicht van de fasen van het project, inclusief de verwachte duur, doelstellingen per fase en de benodigde middelen. De planning is verdeeld in zes fasen, waarbij elke fase zich richt op een specifiek aspect van de ontwikkeling en implementatie van het forumplatform.

in(forum)

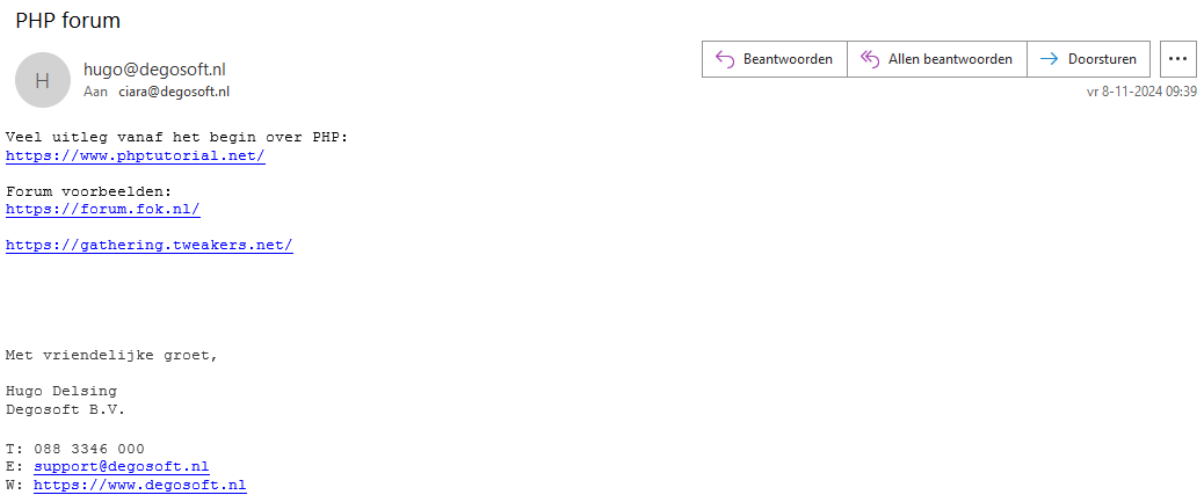
Tabel + Nieuw

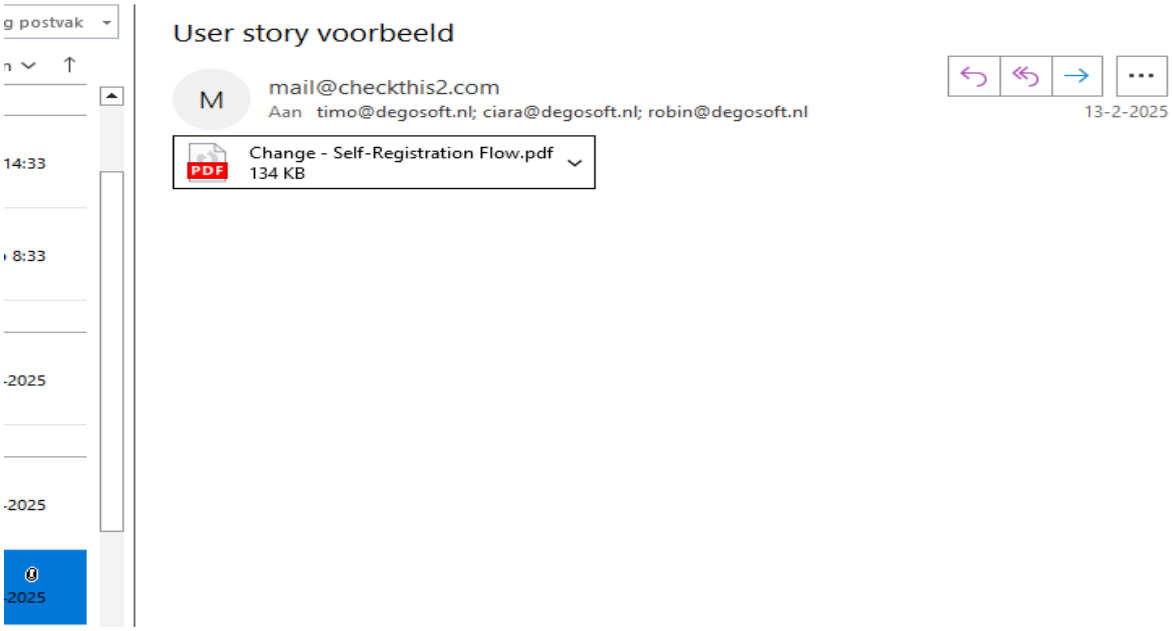
Fase	Duur	Omschrijving	Status
Fase 1: Initiële planning	1 week	Kick-off, basisstructuur, en eerste ontwerp	Gereed
Database ontwerp	Doorlopend	Database modelleren en testen	In uitvoering
Fase 2: Technische opzet	4 weken	Backend implementatie, registratie en login	Gereed
Fase 3: Functionele implementatie	4 weken	Profielen, berichten, en reacties + testen	Gereed
Fase 4: Integratie en testen	2 weken	Integratie van functionaliteiten, feedback verzamelen	In uitvoering
Fase 5: Beveiliging en optimalisatie	2 weken	Beveiliging en prestatie-optimalisatie	In uitvoering
Fase 6: Oplevering en documentatie	3 dagen	Finaliseren en presenteren van het project	Niet gestart
Gebruikerstesten	Doorlopend	Testen met echte gebruikers, feedback	In uitvoering
Training en overdracht	1 week	Documentatie, training, support	Niet gestart

+ Nieuwe pagina

2.7 Contact

Tijdens het project heb ik deelgenomen aan bepaalde overlegmomenten. Ik stelde vragen om de besproken problemen beter te begrijpen. Ik stemde af en toe en tijdig af met mijn teamleden en opdrachtgever over de voortgang en eventuele knelpunten. Alle afspraken werden zorgvuldig vastgelegd en ik hield me hier consequent aan, wat bijdroeg aan een efficiënte samenwerking en het behalen van projectdoelen.





3. Technisch Ontwerp

3.1 Systeemarchitectuur

- Welke technologieën worden gebruikt (frameworks, databases, programmeertalen)

De uitgekozen tool: **Phpmyadmin SQL**

- De HTML-pagina's zullen statisch moeten worden geladen en omgezet naar PHP. En dat de informatie daadwerkelijk uit de database komt. Ook dat de formulieren werken en informaties kunnen worden toegevoegd.

Technologieën:

- Backend: **PHP**
- Database: **MySQL**
- Front-end: **HTML, CSS, Javascript**

- Hoe wordt de dataconsistentie gewaarborgd?

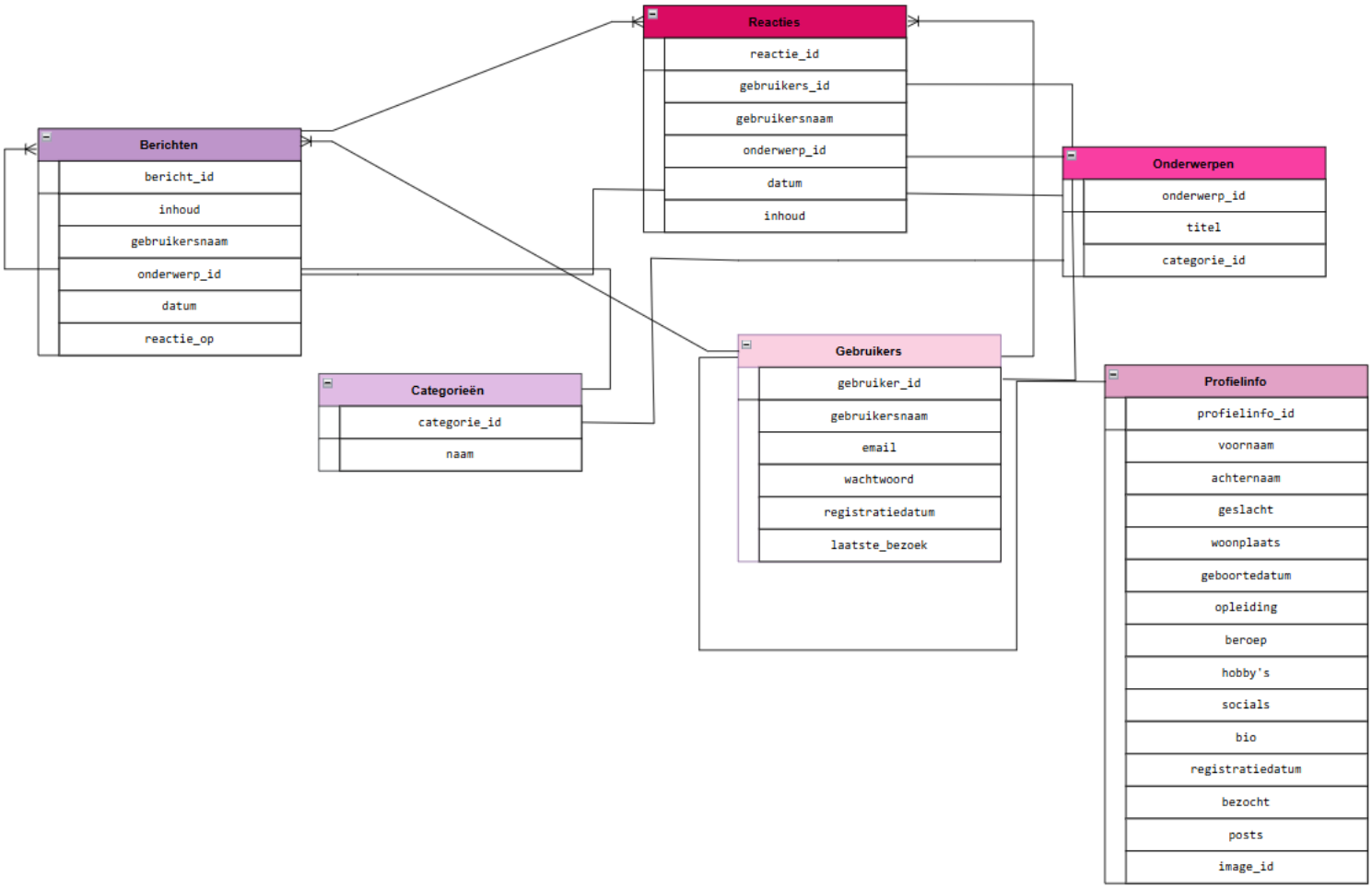
  - N.V.T

3.2 Databaseontwerp

- Welke databasestructuur wordt gehanteerd?

- MySQL-database met tabellen voor bijv.:
  - **Gebruikers** (ID, naam, e-mail, wachtwoord)
  - **Berichten** (ID, titel, inhoud, auteur, timestamp)
  - Reacties (ID, bericht-ID, auteur, inhoud, timestamp)

- Wat zijn de belangrijkste tabellen en relaties?



3.3 Beveiliging en authenticatie

- Hoe wordt de authenticatie en autorisatie geregeld?

  - Gebruikers moeten inloggen via **beveiligde sessies**.
  - Wachtwoorden worden **gehasht** met **bcrypt**.
  - Gebuikers kunnen geen gegevens veranderen die al vaststaan

- Welke beveiligingsmaatregelen worden getroffen tegen aanvallen (bijv. SQL-injectie, XSS)

  - SQL-injectiepreventie

  - Welke onderdelen hebben extra maatregelen gekregen zoals bescherming of afhandeling op acties

- E-mailadres of wachtwoord wijzigen
- Inloggen of registreren
- Formulieren, instellingen of profiel wijzigen
- Profielfoto uploaden
- Account verwijderen
  Instellingen pagina in zijn geheel
- Database beveiliging
- URL-afhandeling

categorie.php

```php
<?php
session_start();
require_once('config.php');


    if (isset($_GET['id']) && is_numeric($_GET['id'])) {
    $categorie_id = intval($_GET['id']);

    $stmt = $conn->prepare("SELECT * FROM categorie WHERE id = ?");
    $stmt->bind_param("i", $categorie_id);
    $stmt->execute();
    $categorie_result = $stmt->get_result();

        if ($categorie_result->num_rows > 0) {
        $categorie = $categorie_result->fetch_assoc();
        $categorie_naam = htmlspecialchars($categorie['naam']);
        } else {
        die("Categorie niet gevonden.");
        }

    // Haal de onderwerpen in de categorie op
    $stmt = $conn->prepare("SELECT * FROM onderwerpen WHERE categorie_id = ?");
    $stmt->bind_param("i", $categorie_id);
    $stmt->execute();
    $onderwerpen_result = $stmt->get_result();
    } else {
    die("Ongeldig categorie-ID.");
    }
?>

```

```
10  // Haal de gebruikersnaam en anders stel het in voor ingelogde gebruikers anders zijn ze anoniem
11  if (!isset($_SESSION['gebruikersnaam'])) {
12
13  $stmt_naam = $conn->prepare("SELECT gebruikersnaam FROM gebruikers WHERE id = ?");
14  $stmt_naam ->bind_param("i", $id);
15  $stmt_naam ->execute();
16  $result_naam = $stmt_naam ->get_result();
17
18  if ($result_naam->num_rows > 0) {
19  $row_naam = $result_naam->fetch_assoc();
20  $_SESSION['gebruikersnaam'] = $row_naam['gebruikersnaam'];
21  } else
22
23  $_SESSION['gebruikersnaam'] = 'Anoniem';
24
25  }
26
27  $gebruikersnaam = $_SESSION['gebruikersnaam'];
28
29  if (isset($_GET['id']) && is_numeric($_GET['id'])) {
30  $onderwerp_id = intval($_GET['id']);
31
32  // Haal onderwerp op
33  $stmt_ond = $conn->prepare("SELECT * FROM onderwerpen WHERE id = ?");
34  $stmt_ond->bind_param("i", $onderwerp_id);
35  $stmt_ond->execute();
36  $result_ond = $stmt_ond->get_result();
37
38  if ($result_ond->num_rows > 0) {
39  $onderwerp = $result_ond->fetch_assoc();
40  } else {
41  die("Onderwerp niet gevonden.");
42  }
43  $stmt_ond->close();
44
45  // Zet de berichten in de database en plaats het bericht bij de juiste gebruiker en chat
46  $stmt_berichten  = $conn->prepare("SELECT * FROM berichten WHERE onderwerp_id = ? ORDER BY datum ASC");
47  $stmt_berichten ->bind_param("i", $onderwerp_id);
48  $stmt_berichten ->execute();
49  $result_berichten = $stmt_berichten ->get_result();
50  $stmt_berichten ->close();
51  } else {
52  die("Geen onderwerp geselecteerd.");
53  }
54
55  if ($_SERVER['REQUEST_METHOD'] === 'POST' && !empty($_POST['inhoud']) && $onderwerp_id > 0) {
56  $inhoud = ($_POST['inhoud']);
57
58
59
60  $gebruikersnaam = isset($_SESSION['gebruikersnaam']) ? $_SESSION['gebruikersnaam'] : 'Anoniem';
61
62  $stmt_post = $conn->prepare("INSERT INTO berichten (inhoud, onderwerp_id, gebruikersnaam, datum) VALUES ( ?, ?, ?, NOW
63  $stmt_post ->bind_param("sis", $inhoud, $onderwerp_id, $gebruikersnaam);
64  $stmt_post->execute();
```

- Hoe wordt dataversleuteling toegepast?

- Wachtwoorden worden gehasht met bcrypt

onderwerp.php | onderwerp.css | reactie.php | profiel.php | registratie.php

registratie.php

```php
<?php

include 'config.php';
session_start();
include 'header.php';

$error='';

if ($_SERVER["REQUEST_METHOD"] == "POST") {
$naam = $_POST['naam'];
$gebruikersnaam = $_POST['gebruikersnaam'];
$email = $_POST['email'];

//Hier word het wachtwoord gehasht in de database en is het niet te zien voor anderen

$wachtwoord = password_hash($_POST['wachtwoord'], PASSWORD_BCRYPT);
// Hier worden de gegevens in de database verwerkt en opgeslagen
$sql = "INSERT INTO gebruikers (naam, gebruikersnaam, email, wachtwoord) VALUES (?, ?, ?, ?)";
$stmt = $conn->prepare($sql);
$stmt->bind_param("ssss", $naam, $gebruikersnaam, $email, $wachtwoord);



if ($stmt->execute()) {
echo "Registratie succesvol gelukt!";
header("Location: login.php");
exit();
} else {
echo "Fout bij registratie: " . $conn->error;

}
$stmt->close();
$conn->close();
}

```

Privacy:

- Het forum voldoet aan de AVG (Algemene Verordening
- Er wordt expliciet toestemming gevraagd voor het opslaan van persoonsgegevens tijdens de registratie.
- Gevoelige gegevens zoals wachtwoorden worden niet in platte tekst opgeslagen, maar versleuteld.
- Gebruikers hebben het recht om hun gegevens in te zien, te wijzigen of te verwijderen.
- Persoonsgegevens worden alleen bewaard zolang dit noodzakelijk is voor het doel van het forum

```
<!DOCTYPE html>
<html lang="nl">
<link href="mainforum.css" rel="stylesheet">
<?php include 'header.php'; ?>

<img class="upper" src="fotos/HBD.jpg">

<div class="container">
<?php

require_once('config.php');


// Haal de forumcategorieën op uit de database hier
$query = "SELECT * FROM categorie";
$result = mysqli_query($conn, $query);

while ($row = mysqli_fetch_assoc($result)) {
$categorie_id = $row['id'];
echo "<a href='categorie.php?id=$categorie_id'>" . $row['naam'] . "</a>";
}
?>
```

onderwerp.php | login.php | onderwerp.css | reactie.php | profiel.php

profiel.php

```php
147 if (isset($row['image_id']) && is_numeric($row['image_id'])) {
150     $stmt = $conn->prepare("SELECT filename FROM images WHERE id = ?");
151     $stmt->bind_param("i", $image_id);
152     $stmt->execute();
153     $result = $stmt->get_result();
154
155     if ($result->num_rows > 0) {
156         $row2 = $result->fetch_assoc();
157         $filename = $row2['filename'];
158
159         echo "<img src='fotos/$filename' alt='Afbeelding'>";
160     } else {
161         echo "Geen afbeelding gevonden.";
162     }
163
164 } else {
165     echo "geen ID opgegeven.";
166 }
167
168
169
170
171     $stmt->close();
172
173
174
175
176 ?>
177
178 <br><br><?php if (isset($_FILES["images"])) : ?>
179 <img src="<?php echo $filename; ?>" alt="Image Uploaded ">
180
181 <?php endif; ?>
182 <label for="voornaam">Voornaam:
183 <?php echo nl2br(htmlspecialchars($row['voornaam'])) ; ?></label>
184 <label for="achternaam">Achternaam:
185 <?php echo nl2br(htmlspecialchars($row['achternaam'])); ?></label>
186 <label for="geslacht">Geslacht:
187 <?php echo nl2br(htmlspecialchars($row['geslacht'])); ?></label>
188 <label for="woonplaats">Woonplaats:
189 <?php echo nl2br(htmlspecialchars($row['woonplaats'])); ?></label>
190 <label for="geboortedatum">Geboortedatum:
191 <?php echo nl2br(htmlspecialchars($row['geboortedatum'])); ?></label>
192 <label for="opleiding">Opleiding:
193 <?php echo nl2br(htmlspecialchars($row['opleiding'])); ?></label>
194 <label for="beroep">Beroep:
195 <?php echo nl2br(htmlspecialchars($row['beroep'])); ?></label>
196 <label for="hobbies">Hobby's/Interesses:
197 <?php echo nl2br(htmlspecialchars($row['hobbies'])); ?></label>
198 <label for="registratiedatum">Registratiedatum:
199 <?php echo nl2br(htmlspecialchars($row['registratiedatum'])); ?></label>
200 <label for="bezocht">Laatst bezocht op:
201 <?php echo nl2br(htmlspecialchars($row['bezocht'])); ?></label>
202 <label for="posts">Aantal posts geplaatst:
203 <?php echo nl2br(htmlspecialchars($row['posts'])); ?></label>
204 <label for="socials">Socials:
205 <?php echo nl2br(htmlspecialchars($row['socials'])); ?></label>
206 <label for="bio">Bio:
207 <?php echo nl2br(htmlspecialchars($row['bio'])); ?> </label>
208
```

Beveiliging:

- Wachtwoorden worden gehasht met bcrypt voor veilige opslag.
- SQL-injecties worden voorkomen door prepared statements te gebruiken in de databasequeries.
- Sessiebeheer wordt beveiligd door het gebruik van veilige cookies en sessie time-outs.

onderwerp.php | login.php | onderwerp.css | reactie.php | profiel.php

login.php

```php
<?php

include 'header.php';
require_once 'config.php';
if (session_status() === PHP_SESSION_NONE) {
session_start();}
session_regenerate_id(true);


$_SESSION['gebruikersnaam'] =  $gebruikersnaam;

    if ($_SERVER["REQUEST_METHOD"] === "POST") {
    if (!empty($_POST['email']) && !empty($_POST['wachtwoord'])) {

    $email = trim($_POST['email']);
    $wachtwoord = trim($_POST['wachtwoord']);


    $sql = "SELECT * FROM gebruikers WHERE email = ?";
    $stmt = $conn->prepare($sql);

    if ($stmt) {
    $stmt->bind_param("s", $email);
    $stmt->execute();
    $result = $stmt->get_result();


    if ($result->num_rows > 0) {
    $row = $result->fetch_assoc();

    // er word bevestigd of het wachtwoord klopt en halt het gehashte wachtwoord uit de database
    if (password_verify($wachtwoord, $row['wachtwoord'])) {

    $_SESSION['gebruikers_id'] = $row['id'];
    $_SESSION['loggedin'] = TRUE;
    $_SESSION['id'] = $gebruiker['id'];
    $_SESSION['gebruikersnaam'] = $gebruiker['gebruikersnaam'];


    // Als het klopt ga naar profiel
    header("Location: profiel.php");
    exit();
    }
    //Anders
    else {
    $error_message = "Ongeldig emailadres of wachtwoord!";
    }

        $stmt->close();
        } else {
        $error_message = "Fout bij het voorbereiden van de query.";
        }
        } else {
        $error_message = "Vul alle velden in.";
        }

        $conn->close();
        }}
```

Prestatie- en beschikbaarheidseisen

- Welke uptime-eisen gelden er voor het systeem?
    - 80%
- Hoe wordt de systeemprestatie gemonitord en geoptimaliseerd

- Bijvoorbeeld/Voorstel: Gebruik van **Google Analytics** of **New Relic** om te meten hoe lang pagina's laden en fouten opsporen.

- Hoe wordt de belasting verdeeld (bijv. load balancing)

- Bijvoorbeeld: Grote websites gebruiken **load balancers** om het verkeer te verdelen over meerdere servers, zodat het systeem niet vastloopt.

3.5 Deployment en onderhoud

De directe deployment strategie: Bij deze strategie worden wijzigingen direct in de productieomgeving geïmplementeerd. Het is een snelle maar risicovolle methode, omdat eventuele bugs of fouten onmiddellijk zichtbaar zijn voor de gebruikers.

- Welke infrastructuur wordt gebruikt
    - Bijvoorbeeld: XAMPP, MySQL en dan de code naar Github
- Hoe wordt logging en foutafhandeling ingericht
    - Bijvoorbeeld: **Logbestanden** (zoals error logs in PHP)
- Wat is het onderhoudsplan en de verwachte levenscyclus van het systeem
    - Bijvoorbeeld: Een forumsoftware kan elk jaar een grote update krijgen, terwijl een kortlopend project misschien geen onderhoud nodig heeft.
    - Dit geval → nieuwe features indien nodig

4 Testplan

- Wat is het doel van het testplan?

Het testplan heeft als doel de functionaliteit, prestaties, beveiliging en gebruiksvriendelijkheid van het forum te valideren. Het plan omvat testscenario's die gerelateerd zijn aan de user stories, inclusief de verwachte resultaten en alternatieve paden.

- Welke testmethoden worden gebruikt?
  - Functionele testen
  - Integratietesten
  - Gebruikerstesten
  - Beveiligingstesten
- Hoe wordt de testomgeving ingericht?
  - Lokale XAMPP-server
  - Testdatabase met testgebruikers
  - Gebruik van testdata

Test Case ID	TC001
Beschrijving	Kan de gebruiker registreren?
Voorwaarden	De gebruiker is op de registratiepagina.
Teststappen	1. Voer een geldig e-mailadres in. 2. Voer een geldig wachtwoord in. 3. Klik op de "Registreren" knop.
Testdata	E-mail: annarain@example.com Wachtwoord: hashed
Verwacht resultaat	De gebruiker kan registreren en de gegevens worden opgeslagen in de database, anders geef foutmelding
Werkelijk resultaat	De gebruiker kan registreren en de gegevens worden opgeslagen in de database.
Resultaat (Geslaagd/Mislukt	Geslaagd
Opmerkingen	De kolommen waren eerst apart in de database, maar samenvoegen bleek beter.

Test Case ID	TC002
Beschrijving	Kan de gebruiker inloggen?
Voorwaarden	De gebruiker is op de inlogpagina.
Teststappen	1. Voer een geldig e-mailadres in. 2. Voer een geldig wachtwoord in. 3. Klik op de "inloggen" knop.
Testdata	E-mail: annarain@example.com Wachtwoord: hashed
Verwacht resultaat	Wachtwoorden worden versleuteld opgeslagen (bcrypt) voor veiligheid. Profielinformatie is standaard privé. De gebruiker kan inloggen op verschillende accounts en anoniem deelnemen aan gesprekken, anders stuur de gebruiker naar een andere pagina
Werkelijk resultaat	De gebruiker kan inloggen op verschillende accounts en anoniem deelnemen aan gesprekken.
Resultaat (Geslaagd/Mislukt	Geslaagd
Opmerkingen	-

Test Case ID	TC003
Beschrijving	Kan de gebruiker profiel bekijken?
Voorwaarden	De gebruiker is op de profielpagina.
Teststappen	1. Voer een geldig e-mailadres in. 2. Voer een geldig wachtwoord in. 3. Klik op de "inloggen" knop. 4. Gebruiker wordt naar profielpagina gestuurd als gegevens kloppen,
Testdata	E-mail: annarain@example.com Wachtwoord: hashed
Verwacht resultaat	Gebruiker wordt naar profielpagina gestuurd als gegevens kloppen en kan profiel bewerken.
Werkelijk resultaat	De gebruiker kan inloggen op verschillende accounts en profiel bewerken
Resultaat (Geslaagd/Mislukt	Geslaagd
Opmerkingen	-

Doel: Gebruiker registreert → logt in → bewerkt profiel → plaatst een bericht → logt uit.

5. Verbetervoorstellen

- Welke fouten en problemen zijn geconstateerd?
  - Langzame laadtijden bij grote forums.
  - Beveiligingsrisico's zoals XSS-kwetsbaarheden.
- Hoe kunnen deze problemen worden verholpen?
  - Optimalisatie van de database-queries.
  - Implementeren van extra validatie bij invoervelden.
  - Verbeteren van caching om de laadtijd te verkorten.
- Wat kan er verbeterd worden in het ontwikkelproces?
  - Meer testmomenten inplannen tijdens de ontwikkeling.
  - Betere documentatie bijhouden van wijzigingen en fixes.
  - Regelmatig feedback verzamelen van gebruikers.

Voorbeeld met tabel voor verbeteringen:

Verbeterpunt ID	VP001
Beschrijving	Maak de inlogstatus van de gebruiker duidelijker.
Huidige situatie	Na inloggen is er geen duidelijke bevestiging dat de gebruiker is ingelogd.
Voorgestelde verbetering	Voeg een welkomstbericht toe en toon de gebruikersnaam in de navigatiebalk.
Verwacht resultaat	Gebruikers zien direct dat ze zijn ingelogd en kunnen makkelijker navigeren.
Prioriteit	Laag
Opmerkingen	De kolommen waren eerst apart, maar samenvoegen bleek beter.

Oplevering:

- Het project wordt opgeleverd na afronding van de testfase waarin alle kernfunctionaliteiten zijn gevalideerd.
- De code en documentatie worden geüpload naar een publieke GitHub repository (link: https://github.com/studentciara/project).
- Een gebruikershandleiding wordt meegeleverd voor het administreren van het forum.
- Het systeem wordt geïnstalleerd op een hostingomgeving met XAMPP voor live gebruik.

Reflectie:

- Tijdens het project heb ik veel geleerd over het werken met PHP en MySQL, vooral over het veilig omgaan met gebruikersdata.
- Het implementeren van beveiligingsmaatregelen was een leerpunt; het was uitdagend om SQL-injecties en XSS goed te voorkomen.
- Het opzetten van een gebruikersvriendelijke interface bleek belangrijk voor de gebruikservaring.
  Het werken met versiebeheer via Git heeft het ontwikkelproces overzichtelijk en gestructureerd gemaakt.
- De grootste uitdaging was het combineren van front-end en back-end aspecten in één samenhangend systeem.

6 D.O.D

Hoewel er geen specifieke deadline werd gelegd is het wel belangrijk om te weten wanneer het de **defenition of done** wordt beschouwd.

De prioriteiten gelden voor de primaire functionaliteiten en als die afgerond zijn dan heb ik een werkend forum die voldoet aan de eisen en wensen. Maar er mochten ook eigen features worden toegevoegd die gunstig zouden zijn voor gebruikers. Om er even op terug te komen de meest belangrijkste functionaliteiten waren:

- **Registratie en login**
- **Gebruikersprofielen**
- **Reactiesysteem**

Deze functies werken in principe naar behoren. Op kleine dingetjes en fouten na, het forum bevat alleen de meest relevante functies en zo heb ik ook bepaalde dingen eruit gelaten omdat ze niet heel relevant waren. Maar het is wel afgerond

- Extra features/verbetervoorstellen die ik zelf wilde toevoegen

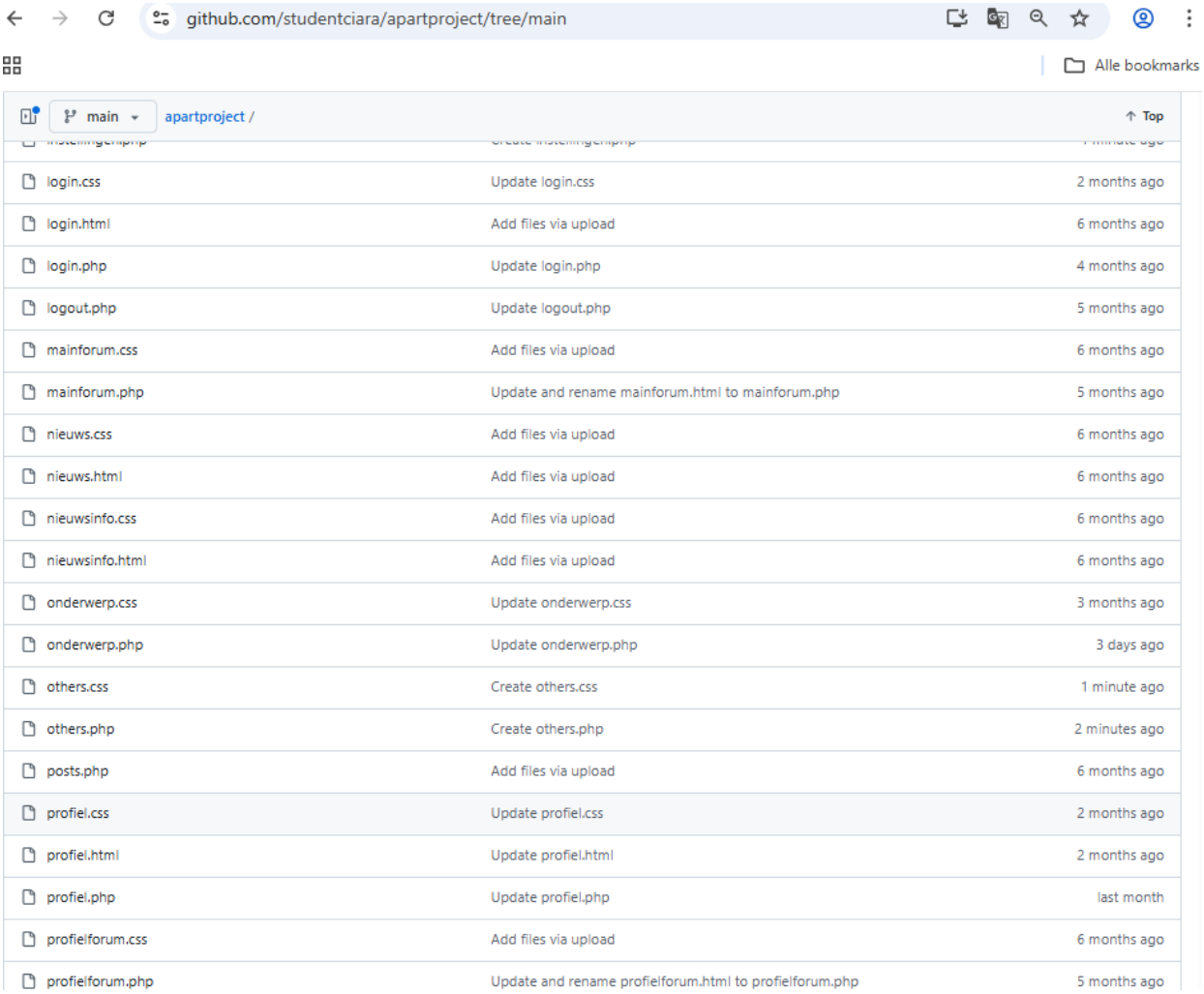
- Notificaties
- Een chat/DM functie die werkt of gewoon met ai
- Het nieuws van andere onderwerpen dynamisch laden
- Badges dynamisch verdienen
-

Hoewel ik deze dingen leuk vond om toe te voegen heb ik ze erbuiten te laten omdat ik niet wist hoeveel tijd ik had maar ik heb ze voor een andere keer gelaten.

Versiebeheer

- Hoe wordt het systeem uitgerold en geüpdatet
    - Versiebeheer bijhouden met nieuwe ideeën
    - **Versiebeheer via Git** (bijvoorbeeld GitHub
    - **Repository**: https://github.com/studentciara/project)
    - **Gebruik van Git flow**: Er worden **feature branches** gebruikt voor elke nieuwe functie, en de code wordt pas gemerged naar de **main branch** na een code review.

Ik heb GitHub gebruikt om mijn voortgang bij te houden. Belangrijke mijlpalen (zoals het toevoegen van de instellingen, postverwerking of profiel aanpassingen) heb ik ge-commit met duidelijke berichten, zodat ik altijd kon teruggaan naar een werkende versie. Zo heb ik geleerd om gestructureerd te werken en fouten sneller op te lossen.



To-Do List:

1. **Projectdoelen & Scope**: Beschrijf de doelen van het project en wat wel/niet wordt gedaan.
2. **Technische Specificaties & Architectuur**: Vermeld gebruikte technologieën en systeemstructuur.
3. **Databaseontwerp**: Geef een overzicht van tabellen en relaties.
4. **UI/UX Beschrijving**: Leg de gebruikersinterface en navigatie uit.
5. **Testplan**: Bepaal welke tests je uitvoert (functioneel, beveiliging, enz.).
6. **Beveiliging & Privacy**: Beschrijf versleuteling en gegevensbeveiliging.
7. **Deploymentplan**: Geef stappen voor implementatie op een live server.
8. **Risicoanalyse**: Identificeer mogelijke risico's en oplossingen.
9. **Opleveringscriteria**: Stel de voorwaarden vast voor projectafronding.

Bronnen

- YouTube (tutorials)
- FOK!Forum
- Tweakers
- W3Schools, CSS-Tricks, CodePen (voor codevoorbeelden)
- Tutorialspoint (tutorials)
- PHP-tutorials (tutorials)
- ChatGPT (uitleg en suggesties)
- Notion (projectplanning)
- GitHub (versiebeheer)
- Pintrest (UI voorbeelden)
- Unsplash (placeholder foto's)
- Google fonts (Icoontjes en typografie)

Bijlagen

In dit document zijn afbeeldingen en screenshots direct bij de bijbehorende onderdelen geplaatst ter ondersteuning van de uitleg. Alle gebruikte afbeeldingen zijn door de auteur zelf gemaakt of gegenereerd binnen de ontwikkelomgeving van het project.

- https://github.com/dashboard