



**MIDDLE BRABANT
LIBRARY**

OVERDRACHTSDOCUMENT

Inhoud

Onderzoeksvraag	3
Eindproducten	3
Communicatieplan	3
Businessplan	3
Web-app	3
Structuur	4
Algemeen	4
Registreren	4
Login/wachtwoord vergeten	4
Home	5
Profile	5
Connecties	5
Chat	5
Uitloggen	5
Technieken	6
Database	6
Account	6
Logged in	6
Interest	7
Messages	7
Garden of Knowledge visual	7
Installatieprocedure	8

Onderzoeksvraag

“How can the library connect Tilburgers with one another in order to share knowledge and reinforce the importance of its role in society?”

Eindproducten

Communicatieplan

Het communicatieplan is het document waarin beschreven staat hoe er naar de doelgroep toe wordt gecommuniceerd. Hier komen vragen aan bod als: Uit welke mensen bestaat de doelgroep?, Wat biedt de Garden of Knowledge aan de doelgroep? en hoe gaan we communiceren naar de doelgroep?

Hierin wordt bepaald welke strategie er wordt toegepast om deze groep aan te spreken en vast te houden en welke middelen daarbij aan te pas komen. De verschillende segmenten komen aan bod en worden toegelicht aan de hand van archetypes. Daarbij hoort ook de inzet van socialmedia om de potentiële gebruikers op de hoogte te brengen van de Garden of Knowledge.

Businessplan

Het businessplan is van belang voor de kosten en business aspect van de Garden of Knowledge. Hierin wordt omschreven wat de missie en visie is, gevolgd door het business canvas model, hierin is o.a neergezet wat de key partners, activiteiten en resources zijn, informatie over de doelgroep en business goals. Verder staat er nog wat de sterktes en zwaktes zijn in de vorm van een swot aangevuld door de planning op korte termijn en lange termijn. Dit gehele document is grondig onderzocht en onderbouwd.

Dit businessplan dient vooral voor advies en onze invalshoek op het project, natuurlijk kunnen zaken veranderen/aangepast worden.

Web-app

De webapplicatie is het persoonlijke onderdeel van the Garden of Knowledge waar gebruikers hun profiel en eigen data kunnen beheren. Het is beveiligd met een login-scherm waarbij de gebruiker met zijn/haar e-mail of bibliotheekpasnummer en een gekozen wachtwoord kunnen inloggen.

Enmaal ingelogd kunnen gebruikers kiezen om hun kennis visualisatie zichtbaar te maken op de smartwall. Daarnaast kunnen gebruikers hun profielgegevens bewerken, een overzicht inzien van actieve gebruikers met gedeelde interesses, berichten sturen naar gebruikers en hun boom visualisatie aanpassen.

Structuur

De webapplicatie bestaat uit een aantal pagina's die de gebruiker daadwerkelijk te zien krijgt en ondersteunende scripts en bestanden in de achtergrond. De indeling per functionaliteit/pagina is als volgt:

Algemeen

- *mysql_connect.php*
dit bestand bevat de vereiste data om te verbinden met de database
- *header_links.php*
dit bestand bevat alle links naar externe bestanden die op iedere scherm in de "<head>" van ieder html scherm worden geladen. Het is gecentraliseerd om consequent hergebruikt te kunnen worden.
- *nav_footer.php*
bevat de html structuur voor de navigatiebalk. Het is gecentraliseerd zodat wijzigingen hierin op iedere scherm consequent doorgevoerd wordt.
- *css*
Bevat style.css de stylesheet met alle vormgeving
- *IMG*
Bevat gebruikte afbeeldingen voor de webapplicatie
 - *IMG/user_pfs*
Bevat geüploade gebruikers profielfoto's

Registreren

- *aanmelden.php*
scherm met formulier om een profiel aan te maken
- *register.php*
voert alle ingevoerde data in de 'account' tabel als nieuwe account;

Login/wachtwoord vergeten

- *login.php*
login scherm met login formulier en wachtwoord vergeten formulier
- *check_login.php*
controleert of de ingevoerde login gegevens overeenkomen met een account en voegt dit account toe aan het 'logged_in' tabel, met een willekeurig start-coördinaat en kleur
- *reset_password.php*
controleert of de ingevoerde id of e-mail overeenkomt met een account en maakt een unieke reset token hiervoor aan die 24 uur geldig is.
- *smtp_iforgot_mail.php*
stuurt een opgemaakte e-mail naar de gebruiker met een resetknop/-link naar nieuwwachtwoord.php
- *nieuw-wachtwoord.php*
reset wachtwoord scherm waar een gebruiker een nieuw wachtwoord kan opslaan. Hiervoor moet de reset link wel geldig zijn. Allereerst worden verlopen tokens uit de database verwijderd en vervolgens wordt de meegegeven token gecontroleerd op geldigheid. Bij een ongeldige token verschijnt een foutmelding met verdere instructies.
- *save_new_password.php*
slaat het nieuwe wachtwoord van een gebruiker op en verwijdert de reset token

- **PHPMailer**
een externe library die het versturen van opgemaakte e-mails vanaf een gegeven e-mailaccount mogelijk maakt.

Home

- *home.php*
home screen van de app, hier komt de gebruiker na een succesvolle login en kan hij/zij zijn/haar boom zichtbaar maken
- *show_on_screen.php*
Maakt een gebruikers boom zichtbaar door in de 'logged_in' tabel 'visibility' op 1 te zetten

Profile

- *profile.php*
Profiel overzichtsscherm waar gebruikers hun persoonlijke profiel kunnen aanpassen en interesses kunnen wijzigen
- *update_profile.php*
Handelt het aanpassen van profielgegevens af

Connecties

- *connections.php*
Scherm met overzicht van actieve gebruiker met gedeelde interesses. Een gebruiker kan de gedeelde interesses bekijken en een andere gebruiker een bericht sturen
- *connection_overview.php*
Haalt alle actieve gebruikers op uit de 'logged_in' tabel die gedeelde interesses heeft en stuurt de data als uitleesbare JSON terug naar connections.php

Chat

- *chat.php*
Scherm met overzicht van berichtgeschiedenis per gebruiker. Door op een van deze gebruikers te klikken opent de berichten thread inclusief invoerveld
- *chat_overview.php*
Haalt alle berichten op uit 'messages' die verstuurd of ontvangen zijn door de ingelogde gebruiker en sorteert deze in op chronologische volgorde in threads gesorteerd per de andere gebruiker
- *new_chat.php*
Indien er voor het eerste naar een gebruiker een bericht wordt verstuurd wordt een nieuwe thread aangemaakt.
- *post_msg.php*
Slaat een verzonden bericht op in de 'messages' tabel en roept getthread.php aan
- *getthead.php*
Ververst de thread inclusief nieuwe berichten en stuurt deze terug naar chat.php. Dit voorkomt dat chat.php ververst moet worden om nieuwe berichten te tonen.

Boom visual

- *tree.php*

Scherf met overzicht van actieve gebruiker met gedeelde interesses. Een gebruiker kan de gedeelde interesses bekijken en een andere gebruiker een bericht sturen

- *get_tree.php*
Haalt alle actieve gebruikers op uit de 'logged_in' tabel die gedeelde interesses heeft en stuurt de data als uitleesbare JSON terug naar connections.php

Uitloggen

- *logout.php*
verwijderd de gebruiker uit de login tabel en beëindigd de sessie.

Technieken

De applicatie is gemaakt in HTML en maakt gebruik van PHP en Javascript om data vanuit een database (MySQL) in te laden in de applicatie.

Database

Binnen de database zijn er een aantal tabellen met data die uitgelezen kunnen worden, deze zijn als volgt:

Account

In dit tabel worden de accounts per gebruiker opgeslagen. Per gebruiker worden de volgende gegevens bijgehouden

- id (unieke gebruikers id, gelijk aan de bibliotheekpas nummer)
- email
- password
- voornaam
- achternaam
- beroep
- interesses
- aangemaakt_op (datum waarop gebruiker is geregistreerd)
- reset token (indien de gebruiker een nieuw wachtwoord aanvraagt wordt een unieke token opgeslagen)
- reset_token_expiry (een reset token is altijd tot 24 uur na aanvraag geldig, hierna wordt hij verwijderd uit de database)

Logged in

In dit tabel staan de gebruikers die ingelogd zijn. Een ingelogde gebruiker wordt voorzien van een willekeurig x en y coördinaat op de smartwall (deze is altijd aan een van de randen) en een willekeurige kleur voor zijn/haar coördinaat. Hierin staan de volgende gegevens:

- user_id (id uit account van de ingelogde gebruiker)
- rand_colour (een willekeurig hex-code voor de takken van de ingelogde gebruiker)
- screen_x (x-coördinaat voor op de smartwall, deze vertaalt zich naar een percentage van de totale breedte)
- screen_y (y-coördinaat voor op de smartwall, deze vertaalt zich naar een percentage van de totale hoogte)
- expiry (een gebruiker blijft normaal een uur ingelogd)
- visible (een gebruiker is per definitie niet zichtbaar wat zich vertaalt in waarde 0, tot zij zich zichtbaar maken op het scherm, waarde 1)

Interest

In dit tabel wordt de volledige interesselijst bijgehouden, de gebruiker kan interesses kiezen die in deze lijst staan, de gekozen interesses worden vervolgens doorgestuurd naar de Account tabel. Deze lijst met interesses kan, indien nodig, aangevuld worden met meer interesses. Ieder interesse heeft een vast coördinaat en deze is uniek voor dit interesse

De volgende data staan hierin:

- id (unieke id van het interesse)
- label (naam/label van het interesse)
- x (x-coördinaat op het scherm, combinatie x en y is uniek voor ieder interesse)
- y (y-coördinaat op het scherm, combinatie x en y is uniek voor ieder interesse)

Messages

Dit tabel bevat de berichten die tussen gebruikers worden verstuurd. **Belangrijk:** dit dient slechts ter demonstratie van een chat en is allerm minst veilig of aan te raden. Berichten worden hier onversleuteld opgeslagen. Voor werkelijke implementatie is het ontwikkelen of integreren van een veilige versleutelde chat server aan te raden.

Hierin staat het volgende:

- message_id (unieke id van het bericht)
- sent_from (gebruikers id van de afzender)
- sent_to (gebruikers id van de ontvanger)
- message (het daadwerkelijke bericht, opgeslagen als "plain text")
- time (datum en tijd waarop bericht verstuurd is)

Garden of Knowledge visual

De garden of knowledge is de visual die op de smartwall in de LocHal verschijnt. De visual bestaat uit coördinaten van interesses van mensen die ingelogd zijn in de web-app. Naar die coördinaten toe lopen lijnen vanuit de zijkant van het screen. Aan de zijkant van het screen zijn de begin coördinaten van ingelogde gebruikers. Beide coördinaten worden opgehaald uit de database. Om het visueel aantrekkelijk te maken is er gebruik gemaakt van bezier-curves om de lijnen op wortels te laten lijken. Hieronder staan de formules voor het gebruik van een bezier-curve. Het gebruik van deze curve is niet een vereiste, maar maakt het geheel wel visueel aantrekkelijker.

$$x = (1-t)^2 * 0 + 2(1-t)t * 0.5 + t^2 * 1 = (1-t)t + t^2 = t$$

$$y = (1-t)^2 * 0 + 2(1-t)t * 1 + t^2 * 0 = 2(1-t)t = -t^2 + 2t$$

Dit deel van de applicatie is gebouwd in html met javascript ondersteuning. Gebruikte libraries zijn JQuery met de snap.svg plugin. Om de bezier-curve succesvol om te zetten in javascript is de volgende formule gebruikt:

path d="M X₁, Y₁, C dX₁, dY₁, dX₂, dY₂, X₂, Y₂"

- X₁ = begin coördinaat x-as (de gegeven x-as coördinaat voor gebruiker die inlogt)
- Y₂ = begin coördinaat y-as (de gegeven y-as coördinaat voor gebruiker die inlogt)
- dX₁ = controlepunt 1 x-as (random x-as coördinaat om een straal om de x₁ heen)
- dY₁ = controlepunt 1 y-as (random y-as coördinaat om een straal om de y₁ heen)
- dX₂ = controlepunt 2 x-as (random x-as coördinaat om een straal om de x₂ heen)
- dY₂ = controlepunt 2 y-as (random y-as coördinaat om een straal om de y₂ heen)
- X₂ = eind coördinaat x-as (de gegeven x-as coördinaat van een interesse)
- Y₂ = eind coördinaat y-as (de gegeven y-as coördinaat van een interesse)

Structuur

Visual

- *index.php*
Scherm met de visual die daadwerkelijk getoond wordt.

Snap svg

Voor de plugin snap.svg is eerst de javascript library genaamd JQuery nodig. Deze is gelinkt vanuit google apis en kan te allen tijde via deze link gelinkt worden in de <head> van de html:

<https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js>

In de "js" map in de hoofdmap zit het volgende bestand dat nodig is om snap.svg te kunnen gebruiken:

- *snap.svg-min.js*
Een compressed plugin bestand van snap.svg

Installatieprocedure

Om de Garden of knowledge technisch werkend te krijgen, is het belangrijk om de volgende stappen te doorlopen:

- Overzetten van de database
De database is essentieel in het project, data moet natuurlijk ook opgeslagen kunnen worden. Het aangeleverde SQL bevat de gehele benodigde database, die vervolgens gemakkelijk geïmporteerd kan worden naar de eigen database. Het is noodzakelijk dat deze in orde is voordat de volgende stap te pas komt.
- Overzetten naar eigen webhost
Net als de database is dit ook essentieel in het project. Zonder webhost is er geen webpagina. Belangrijk is om alle bestanden naar de eigen webhost over zetten. Hierbij kan gebruik worden gemaakt van Filezilla. Wanneer dit met succes is voltooid, kan de site worden bereikt via de gekozen naam bijvoorbeeld: "www.gardenofknowledge.nl".

Het is voor de bieb belangrijk als de naam van de site binnen de domein van de bieb blijft. Dit om ervoor te zorgen dat gebruikers de link naar de bieb kunnen leggen en er geen extra kosten bijkomen voor het kopen van een nieuwe domeinnaam.

- MySQL login data aanpassen
De logingegevens van de MySQL zijn in dit stadium misschien incorrect, deze zullen aangepast moeten worden in '*mysql_connect.php*'.
- FTP login data aanpassen
De logingegevens van de ftp server (voor het uploaden van profielfoto's zijn in dit stadium misschien incorrect, deze zullen aangepast moeten worden in '*submit_pf.php*'.
- Communicatie in gang zetten
De volledige communicatiestrategie staat in het communicatieplan. Lees dit document goed voordat de communicatie in gang wordt gezet.

Recommendations

Na aanleiding van het project zijn er enkele zaken die wellicht interessant zijn die helaas niet aan bod zijn gekomen tijdens het project vanwege, tijdgebrek of een soort van gelijke reden. Deze aanbevelingen zijn geheel vrijblijvend natuurlijk, maar wel natuurlijk aangeraden. De aanbevelingen zijn gebaseerd op voorlopig onderzoek en invloed van experts, om ervoor te zorgen dat de aanbevelingen grondig beredeneerd zijn.

Profiel foto

Een profielfoto is momenteel niet van toepassing. Echter dit is wel een toevoeging voor de web-app, aangezien hierdoor mensen elkaar beter kunnen herkennen en geeft een wat vriendelijkere sfeer. Natuurlijk moet de optie voor een profielfoto optioneel worden en mensen zouden er ook voor moeten kunnen kiezen om geen profielfoto te nemen.

Chatfunctie

De huidige chat functie is enkel een prototype en wordt d.m.v een database geregeld, dit is niet optimaal en de verschillende chats zullen snel de database overstromen zou het in de huidige staat implementeren. The implementatie van een chatfunctie in koppeling natuurlijk met een chatserver zullen ervoor zorgen dat de chatfunctie optimaal werkt en zonder de database te onderstromen met alle berichten die worden verstuurd. Ook voor de privacy van de gebruiker zal dit beter zijn. Hiervoor zijn enkele oplossingen mogelijk maar deze hangen af van verschillende factoren:

- Prijs, sommige zijn gratis maar zullen niet zo uitgebreid/customizable zijn.
- Moeilijkheidsgraad van integratie, het is belangrijk dat de bibliotheek ook in staat is om het te integreren.
- Customizable, het is van belang dat de chatfunctie goed werkt maar ook dat het goed in de stijl van de web-app passen.

Het is te adviseren om een simpel te integreren chatfunctie te kiezen of een programmeur of expert inhuren om dit in orde te krijgen of heel de chatfunctie op te zetten en te implementeren. De manier is niet van hoog belang maar wel dat de functie er komt en dat deze een vorm van encryptie (versleuteling) bevat.

Encryptie

De encryptie van de gegevens is momenteel niet optimaal en is zeker open voor verbetering. Encryptie dient er voornamelijk voor dat de gegevens van de gebruikers ge-encrypt worden, dit zorgt ervoor dat de gegevens beschermd zijn en dat iemand die de database zou inkomen niet de gegevens zou kunnen uitlezen. In principe worden de gegevens versleuteld.

End to End encryption is zeer aangeraden, deze vorm van encryptie wordt tevens gebruikt door Whatsapp. Deze encryptie zorgt ervoor dat mensen niet mee kunnen kijken, met mensen moet je eerder denken aan servers die het 'verkeer' monitoren en zo de niet-encrypted info leest. Gebruikers vinden het over het algemeen prettiger wanneer hun privacy wordt beveiligd.

Customisatie

Customisatie is cruciaal aangezien dit de gebruiker de vrijheid geeft om zichzelf te uiten (tot een zekere zin). De gebruiker kan zijn boom/wortels aanpassen naar eigen smaak. Uit een psychologisch onderzoek kwam dat een gebruiker voelt zich dan beter over zichzelf. Verder zorgt de customisatie voor verdieping in de web-app, de gebruiker zal meer tijd spenderen en mocht dit een succes zijn, dan is er ook de mogelijkheid om enkele onderdelen van deze visuele customisatie wat extra's toe te voegen voor een kleine prijs, dit moet puur alleen visueel zijn en geen voordelen hebben vergeleken met niet betalende gebruikers, dit kan slecht uitpakken. Wanneer de gebruikers het gevoel krijgen dat ze voor de web-app moeten betalen zullen vrij veel gebruikers afhaken.

Visuele effecten

In de web-app is er nog ruimte voor enkele visuele effecten die een toevoeging zullen bieden voor de web-app.

Persoonlijke boom:

- Interesse: Momenteel zie je alleen de visualisatie, het is een mogelijkheid om een optie te maken om je interesses aan te klikken en wordt er vervolgens informatie over de interesse weergegeven.

“Smart Wall” visualisatie:

- Connectiepunten: de connectiepunten op de grote visualisatie zijn niet super opvallend, een visueel effect wanneer een connectie wordt gemaakt zou handig zijn. Dit zorgt ervoor dat de connectie duidelijk opvalt.