

Design Rationale



Project Discover

27/8/2022
Emile Kost
Joep Klaassen

Debriefing

Probleem

De aanleiding van deze opdracht is dat veel mensen na het afronden van hun opleiding niet een duidelijk beeld hebben van een baan die zij zoeken en wat zij zelf willen. Dit zorgt vaak voor ontevreden werknemers. Ze zijn niet tevreden met hun baan, maar weten niet waar ze wel willen werken.

Doel

De doelstelling van dit project is om gebruikers een passende baan aan bieden. Het project is een vacaturebank met een andere invalshoek dan gewoonlijk. Gebruikers kunnen bladeren door vacatures, en zoeken naar specifieke vacatures. Echter wanneer een gebruiker totaal geen idee heeft waar hij/zij naar opzoek is, of geen specifieke richting op wil is er een extra functionaliteit. Een gebruiker kan dan een persoonlijkheidstest doen, en krijgt aan de hand daarvan relevante vacatures aangeboden, die passen bij zijn karakter.

Gebruiker

De eindgebruikers van de website zijn net afgestudeerde studenten die net beginnen met het zoeken naar een baan. Ook vallen werknemers die niet tevreden zijn met hun huidige baan binnen de gebruikers. De opdrachtgever heeft aangegeven te willen testen binnen de BioTech sector.

Randvoorwaarden

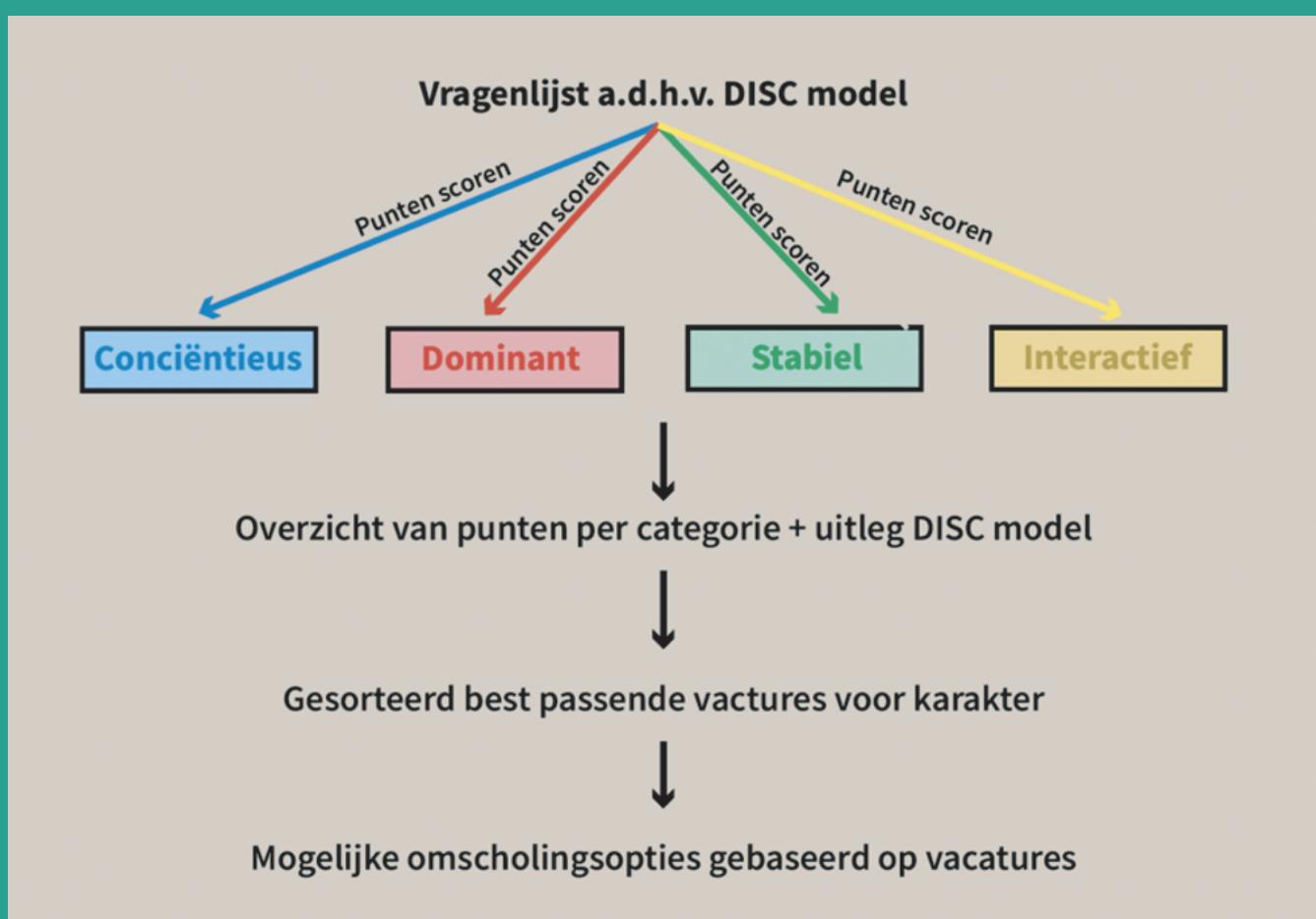
- Aantrekkelijke website waar BioTech studenten zich thuis voelen
- Volledig responsive
- DISC model persoonlijkheidstest geïmplementeerd
- Relevante vacatures a.d.h.v. DISC model

Werking website

Aan de hand van het gesprek met de opdrachtgever hebben we een overzicht gemaakt van hoe de website moet gaan werken. Er zijn natuurlijk de basis functionaliteiten van een vacaturebank, die staan op de volgende pagina verder toegelicht.

Maar er is ook een extra functie, namelijk de DISC test. In het begin was het voor ons een beetje vaag hoe dit in zijn werk zou moeten gaan. Vandaar dat we een overzicht hebben gemaakt met de relaties en werking van deze functionaliteit. De gebruiker begint met een vragenlijst invullen, en krijgt per vraag één of meerdere punten per DISC categorie. Daarmee wordt een score berekent, en aan de hand daarvan passende vacatures aangeboden.

Het DISC model is een gedragsmodel bedacht door Dr. William Moulton. Het is een model met vier fundamentele gedragsstijlen. Het karakter van een mens past ergens in de vier categorieën. Vaak is er een dominante categorie, en de passen mensen voor de rest tussen meerdere categorieën in.



Functionaliteiten

Profiel aanmaken

Gebruikers worden bij hun eerste bezoek van de website meteen doorverwezen naar de login pagina. Het is niet mogelijk om de website te gebruiken zonder account. Hier kan een gebruiker een profiel aanmaken voor een persoonlijke ervaring op de website. Het profiel is ook noodzakelijk om andere data van de website op te slaan, voor volgende bezoeken van de gebruiker.

DISCOVER

Registreer jouw account

Naam
Vul hier jouw naam in

E-mail
Vul hier jouw email in

Wachtwoord
Vul hier jouw wachtwoord in

Account aanmaken

Al een account? Log hier in

Login

Inloggen

Als een gebruiker de website al eerder heeft bezocht kan hij inloggen met zijn account. Wanneer je niet uitlogt blijf je automatisch op je eigen account de website gebruiken bij latere bezoeken.

DISCOVER

Login met jouw account

E-mail
Vul hier jouw email in

Wachtwoord
Vul hier jouw wachtwoord in

Inloggen

Nog geen account?
Account aanmaken

Vacatures bekijken

Op de homepagina van de website kan de gebruiker uitgelichte vacatures bekijken. Hier staan een aantal vacatures onder elkaar. Ook zit er bij elke vacature een detailpagina met extra informatie, en een button om te solliciteren, en op te slaan in favorieten.

DISCOVER

Profiel Home Zoek naar vacatures... Zoeken

Welkom terug Joep

Waarom de DISC test? Bekijk vacatures, of zoek in de zoektafel naar specifieke vacatures.

Weet je nog niet waar je naar op zoek bent? Doe de DISC test! Een persoonlijkheidstest waar karaktereigenschappen uitkomt, zodat je vacatures kan bekijken die bij jou passen.

DISC persoonlijkheidstest

Uitgelichte vacatures

Biofarmaceutisch Operator

Ben jij toe aan een nieuwe baan in een innovatief farmaceutisch bedrijf? Dan is deze vacature als farmaceutisch operator wat voor je! Bij dit bedrijf werk je allerlei soorten bio-farmaceutische producten in een gezellige omgeving.

- Amsterdam, Nederland
Zoekvacature

Technician Bioprocessing

Ben jij toe aan een nieuwe baan in een innovatief farmaceutisch bedrijf? Dan is deze vacature als farmaceutisch operator wat voor je! Bij dit bedrijf werk je allerlei soorten bio-farmaceutische producten in een gezellige omgeving

- Amsterdam, Nederland
Pharmaceutical , Biotechnology, Medical Device & Diagnostics

Functionaliteiten

Vacatures zoeken

Boven in de header staat een zoekfunctie waarmee de gebruiker kan zoeken naar vacatures. Dit is erg handig voor gebruikers die al weten waar ze naar opzoek zijn.

DISCOVER Profiel Home Zoek naar vacatures... Zoeken

Welkom terug Joep

Welkom bij de biotechnische vacaturebank. Bekijk vacatures, of zoek in de zoekvak naar specifieke vacatures.

Wat is DISC? Misschien moet waar je hebt op zoek gestaan. Doe je DISC test! Een persoonlijkheidstest om karaktereigenschappen te ontdekken, zodat je vacatures kan bekijken die bij jou passen.

DISC persoonlijkheidstest

Uitgelichte vacatures

Assistant quality manager (4-tal maanden)

Ditso kent, gelegen tegenover, is op zoek naar een assistent quality manager. Ze zoeken iemand die bij voorkeur al enige ervaring heeft met product specificaties. <input type="checkbox"/> Wie ben ik?

- Amsterdam, Netherlands
- Life Sciences, Materials Science

Solliciteren

Vacatures toevoegen aan favorieten

Als een gebruiker een vacature wilt opslaan om later te bekijken/onthouden kan hij deze toevoegen aan favorieten. Dit wordt dan verstuurt naar de database zodat het opgeslagen staat.

DISCOVER Profiel Home Zoek naar vacatures... Zoeken

Verantwoordelijkheden

Deze functie vergt de gedetailleerde procedures te ontwikkelen, namen, indexeren, registreren, opladen, uitvoeren en gereedmaken voor ververging. Verzending en opgaven van documenten en rapporten aan externe partijen. Deelname aan het archiveringsproces. Beheren van het archiefdocumentaire en internationale werkomgeving.

Werkomstandigheden

Duurzaam inkomen tot op scherpe leeftijd. Salaris €2000-€2500 bvv ervaring. Goede secundaire arbeidsvooraarden.

Wie ben jij?

Je hebt een grote verantwoordelijkheidszin

Kernwoorden

- Vriendelijk
- Enthusiast
- Nauwkeurig

Solliciteren

DISC persoonlijkheidstest

Voor gebruiker die nog niet zo goed weten waar ze naar opzoek zijn is de DISC test. Na het doen van deze test krijgt de gebruiker een uitslag van karaktereigenschappen.

DISCOVER Profiel Home Zoek naar vacatures... Zoeken

Personenlijkhedstest

Vul alle vragen in om een beeld te vormen van jouw karaktereigenschappen.

Wat is een directe test?

Ben jij direct of indirect?

- Direct
- Indirect

Ben jij taakgericht of mensgericht?

- Mensgericht
- Taakgericht

Geef wat het beste bij jou past

- Direct
- Betrouwbaar
- Harmonieus
- Analytisch

Vacatures adhv DISC test

Na het doen van de DISC test krijgt de gebruiker resultaten met overeenkomsten van verschillende DISC categorieën. Aan de hand deze resultaten krijgt hij dan relevante vacatures te zien

DISCOVER Profiel Home Zoek naar vacatures... Zoeken

Vacatures op basis van jouw DISC resultaten

Wat is een directe test?

Technician Bioprocessing

Bent u toe aan een nieuwe baan in een innovatieve biomedische industrie? Dan is deze vacature als farmaceutisch operator wellicht wat voor u! Bij dit bedrijf maakt je werk deel uit van de productie en verwerking van producten in een gezonde omgeving

- Amsterdam, Nederland
- Pharmaceutical technology
- Medical Devices & Diagnostics
Afgerond MBO of HBO opleiding in een relevante sector, zoals farmaceutisch bio proces envering onder GMP. Goede communicatie, bij voorkeur ook in het Engels.

Kernwoorden

- Logistiek
- Fysiek
- Productie

Assistant quality manager (4-tal maanden)

Orzo kent, gelegen tegenover, is op zoek naar een assistent quality manager. Ze zoeken iemand die bij voorkeur al enige ervaring heeft met product specificaties. <input type="checkbox"/> Wie ben ik?

- Amsterdam, Netherlands
- Life Sciences, Materials Science

Je hebt een grote verantwoordelijkheidszin

Kernwoorden

- Vriendelijk
- Enthusiast
- Nauwkeurig

Solliciteren

Schema opstellen

Om de data te structureren hebben wij in een Javascript bestand een Schema opgezet. In dit schema's maken we titels aan en willen wij weten om wat voor data het gaat. Dit kan bijvoorbeeld een String, Array of Boolean zijn.

Dit datamodel wordt later weer geexporteerd naar het app.js bestand zodat de data daadwerkelijk erin kan worden gezet. Omdat de titel het enkelvoud is van de meervoudige collectie naam, weet Mongoose precies om welke collectie het gaat.

Registreren en inloggen met een account

Bij ons concept moet er veel persoonlijke data worden verwerkt. Discover is namelijk een vacaturebank die interactief met gebruikers omgaat en daarom informatie moet kunnen opslaan dat gekoppeld kan worden aan een account. Voor het data opslaan hebben wij gebruik gemaakt van Mongodb database, hierover later meer. Voor het registreren en inlog systeem hebben wij gebruik gemaakt van Passport en Bcrypt. Passport wordt gebruikt voor het relatief gemakkelijk opstellen van een login systeem en Bcrypt wordt gebruikt voor het encrypten van het wachtwoord. De gebruiker kan pas daadwerkelijk toegang krijgen tot ons product als hij een account heeft geregistreerd en succesvol heeft ingelogd.

Registreren van het account

```
app.post('/register', checkNotAuthenticated, async (req, res) => {
  try {
    const hashedPassword = await bcrypt.hash(req.body.password, 10)

    //This is the required data that has to be stored in mongodb
    const blog = new Blog({
      name: req.body.name,
      email: req.body.email,
      password: hashedPassword,
    })

    //Saves user account data in mongodb, password is still secure and encrypted
    blog.save()
      .then((result) => {
        console.log(result)
      })
      .catch((err) => {
        console.log(error)
      })

    res.redirect('/login')
  } catch {
    res.redirect('/register')
  }
})
```

Op de registratie pagina moet de gebruiker zijn email, naam en een zelfbedacht wachtwoord invullen. Deze data wordt door middel van Body-parser uit de formulieren opgehaald. Dit is mogelijk door req.body omdat wij Body-parser hebben gebruikt.

Na het ophalen van de waardes. Wij hebben gebruikt gemaakt van een try statement zodat tijdens het uitvoeren van de code tegelijkertijd naar foutmeldingen wordt gezocht. Bij de variabele user wordt er een nieuw dataobject voor Mongodb aangemaakt. Door middel van .save wordt uiteindelijk de data in de database opgeslagen. Deze informatie zal gekoppeld worden aan een uniek objectId en kan later opnieuw worden opgehaald.

Uitleg code

Mongodb en Mongoose

Om data op te slaan hebben wij voor het eerst met een database gewerkt. Hiervoor hebben wij de cloud based database Atlas van Mongodb gebruikt. Dit hebben wij gecombineerd met Mongoose, zodat er makkelijk gecommuniceerd kan worden tussen de server en de database. In dit kopje gaan we bespreken hoe we hier tot gekomen zijn.

Account en database maken op Mongodb

Na het maken van een account kan je gelijk een gratis database aanmaken. Deze hoef je alleen een titel te geven en voila. In deze database kan je verschillende collecties aanmaken. Deze collecties kan je in een Javascript bestand meegeven hoe je ze gestructureerd wilt hebben en welke data er moet worden opgeslagen. Het is een good practice als de naam van de collectie een meer-voud is van de Schema's die er later in komen te staan.

The screenshot shows the MongoDB Atlas web interface. At the top, there are tabs for 'Atlas', 'App Services', and 'Charts'. Below the tabs, a search bar says 'Search Namespaces'. Underneath, a sidebar shows a database named 'VacaturebankData' with two collections: 'blogs' and 'offers'. The 'blogs' collection is currently selected. The main area displays two documents:

```
_id: ObjectId("62b2f1b69622a15e86fd3c56")
name: "Joep"
email: "joep.klaassen9@gmail.com"
password: "$2b$10$PLrkRGvd57r3uKxu4MueJlBgI6Kslsm5qLhNF7ERmQEA4sFFwW."
favorites: Array
createdAt: 2022-06-22T10:40:54.454+00:00
updatedAt: 2022-06-22T15:23:04.555+00:00
__v: 0

_id: ObjectId("62b40e0562a71e007e72b659")
name: "Johan"
email: "jioeo.klaassen9@mail.com"
```

Connectie krijgen met de database

Om een connectie met de database op te zetten kan je op de website een url aanvragen. In deze url moet je alleen nog je developer gebruikersnaam en wachtwoord toevoegen.

```
mongoose.connect(dbURI, { useNewUrlParser: true, useUnifiedTopology: true })
  .then((result) => app.listen(3500), console.log('Mongodb connected'))
  .catch((error) => console.log(error + 'has occured'))
```

Inloggen

Na het aanmaken van een account wordt de gebruiker doorverwezen naar de login pagina. Hier moet de gebruiker inloggen met zijn email en wachtwoord. Als dit succesvol wordt gedaan, dan wordt in het app.js bestand aangegeven dat de gebruiker moet worden doorverwezen naar de index pagina door middel van succesRedirect: '/'. Als dit niet zo is, dan blijft de gebruiker op de login pagina en kan hij het opnieuw proberen.

```
app.post('/login', checkNotAuthenticated, passport.authenticate('local', {
  successRedirect: '/',
  failureRedirect: '/login',
  failureFlash: true
}))
```

In de app.post van de login pagina wordt hier aangegeven wat moet gebeuren bij het succesvol of onsuccesvol invullen van het login form.

Home/index pagina

Na het succesvol inloggen komt de gebruiker op de indexpagina terecht. Vanuit hier kan die gelijk de DISC test starten of kan die al browsen en zoeken naar de beschikbare vacatures. Ook kan hij hier door navigeren naar andere pagina's. Voordat de gebruiker kan beginnen met het browsen en zoeken naar vacatures moeten deze eerst worden opgehaald vanuit de database.

Ophalen van vacatures

```
const Offers = require('../models/joboffers.js');

app.get('/', checkAuthenticated, async (req, res, next) => {

  Offers.find((err, docs) => {
    if(!err) {
      res.render('index', {
        data: docs,
        name: req.user.name
      })
    } else {
      console.log('Failed to retrieve data')
    }
  })
})
```

Voordat er informatie vanuit de juiste collectie kan worden opgehaald moet eerst het passende datamodel worden gedeclareerd. In dit geval is dit het model jobOffers, omdat we opzoek zijn naar de vacatures. Omdat we alle vacatures willen ophalen kunnen we de methode .find() gebruiken. Hierin geven wij aan dat we errors en alle documenten zoeken. Als er geen foutmelding naar boven komt, dan wordt de indexpagina gerenderd en worden de vacatures in een object meegegeven.

En als er wel een error is dan zal dit in de console.log komen te staan, maar zolang er een connectie is met de database zal dit niet zo snel gebeuren. De functie checkAuthenticated wordt vaak meegegeven. Deze functie zorgt er namelijk voor dat wij gebruikersdata zoals de naam gemakkelijk kunnen opvragen.

Renderen van de vacatures

Voor het gemakkelijk renderen van data vanuit Node.js hebben wij gebruikt gemaakt van ejs. Zodat wij gemakkelijk client side Javascript in de HTML kunnen schrijven. Omdat er voor meerdere vacatures moet en worden gerenderd maken wij hier gebruik van een for loop. Dit had tevens ook met een forEach gekund. Met deze for loop gaan we door de gehele array heen en wordt een voor een elke vacature gerenderd tot dat ze allemaal zijn gerenderd.

Na het opstellen van de for loop kunnen we met [i] in de HTML voor elke vacature de structuur en content bepalen.

De link in de pagina staat gelijk aan het objectId. Vanuit hier kan de gebruiker gemakkelijk door naar de detailpagina.

Zoeken van vacatures op de indexpagina

Op de index pagina heeft de gebruiker naast het browsen van vacatures en het starten van de DISC test ook de optie om te kunnen zoeken in de database naar vacatures. Wij hebben hier gebruik gemaakt van body parser om zo gemakkelijk waarden uit de HTML op te halen. Ook hebben wij Regex gebruikt, wat het mogelijk maakt om snel en gemakkelijk naar data te zoeken in de database.

```
app.post('/', checkAuthenticated, async (req, res) => {
  const input = req.body.search;
  console.log(input);
  let search = await Offers.find({title: {$regex: new RegExp('^' + input + '.*', 'i')}}).exec()
    .then(response => {
      console.log(response)
      res.render('index', {data: response, name: req.user.name})
    })
})
```

Toevoegen aan favorieten

Op de detailpagina krijgt de gebruiker de mogelijkheid om de vacature aan zijn favorieten lijst toe te voegen. Het id van deze vacature wordt aan het user model toegevoegd in de database onder het veld "favorites". In favorites komt een array met objectId's te staan. Deze kunnen later op de profielpagina weer worden gerenderd.

```
app.post('/:id', checkAuthenticated, (req, res) => {
  const user = req.user.id;
  const offerId = req.params.id;

  Blog.findOneAndUpdate({
    _id: user
  }, {
    $push: {
      favorites: offerId
    }
  })
    .then((result) => {
      console.log(result)
      res.redirect('profile')
    })
})
```

DISC Test pagina

De DISC test is een formulier waarbij gebruikers verschillende opties krijgen, zij moeten aangeven welke zij het meeste bij zichzelf vinden passen. Deze data moet worden opgeslagen. Op basis van deze score worden er vacatures op basis van deze score gerenderd.

Opslaan resultaten DISC test

Elke radiobutton heeft bij elke vraag een verschillende value. Deze kan dominant, interactief, stabiel of conscientieus zijn. Bij de submit van het formulier worden al deze waarden opgeslagen bij het juiste kernwoord. Door verschillende if statements worden alle woorden gecontroleerd om te kijken of zij de hoogste score hebben. Het woord met de hoogste score is uiteindelijk het woord waarop de vacatures gebaseerd moeten zijn. De vacatures uit de database hebben al een veld met score waarbij dit uit een van de vier woorden bestaat. Door de .find() methode kan er gezocht worden naar vacatures waarbij het score veld gelijk staan aan het winnende woord. Hierdoor worden alleen deze vacatures gerenderd en zijn deze op basis van de resultaten van de DISC test

Activity diagram

Hieronder staat het activity diagram van de website afgebeeld. Dit is een visuele representatie van hoe de functionaliteiten werken, en welk deel code daar een rol bij speelt.



Verdere uitleg code

De gehele uitleg van alle code staat in de wiki beschreven. In de repository [EmileKost/Discover](#). Daar staat in de wiki een kopje met code uitleg design rationale.

Installeren

Conclusie probleem verholpen omdat...

Zoals in de debriefing is beschreven, is het probleem dat mensen geen duidelijk beeld hebben van wat voor baan ze zoeken. Ze hebben niet een goed beeld van hun karakter, en wat voor baan bij hun past. Dat probleem wordt verholpen door de website die wij hebben gebouwd. Gebruikers met geen goed beeld van hun karakter kunnen de DISC test doen. In de resultaten wordt dan duidelijk in welke categorieën een gebruiker het best past. Dit geeft een idee van wat voor karakter je hebt, en wat bij jouw manier van doen past. Na de resultaten worden relevante vacatures aangeboden aan gebruikers. Hierdoor krijgen gebruikers met geen idee wat ze willen passende suggesties. Gebruikers die wel weten wat ze willen kunnen zoeken en bladeren naar vacatures. De website ondersteunt dus elk soort gebruiker in het vinden van een nieuwe baan.

Hoe kan je de website installeren?

1. Clone de Github repository

<https://github.com/EmileKost/Discover.git>

2. Installeer alle packages

npm install

3. Start de applicatie

npm start

4. Open de website in de browser

localhost:3500