MILAN LEIJSER, RENO ROVERS , HUSSEIN OMAR & JOEY REHMANN

October 2, 2024

Technisch ontwerp

Geoprofs

Contents

[1. Inleiding 3](#_Toc179543969)

[Doel: 3](#_Toc179543970)

[Scope: 3](#_Toc179543971)

[Achtergrondinformatie over het Project 4](#_Toc179543972)

[2. Systeemarchitectuur 5](#_Toc179543973)

[3. Technieken en technologieën 6](#_Toc179543974)

[4. Opslag van data 7](#_Toc179543975)

[ERD 8](#_Toc179543976)

[5. Toegankelijkheid en bereikbaarheid van gegevens 11](#_Toc179543977)

[Data encryptie 13](#_Toc179543978)

[Data in rust 13](#_Toc179543979)

[Data in transport 13](#_Toc179543980)

[Logging 13](#_Toc179543981)

[7. Prestaties en schaalbaarheid 14](#_Toc179543982)

[8. Onderhoudbaarheid 15](#_Toc179543983)

[9. Platforms 16](#_Toc179543984)

# 1. Inleiding

Doel: Het doel van het technisch ontwerp voor het registratiesysteem van GeoProfs is om een efficiënt en gebruiksvriendelijk platform te ontwikkelen voor het indienen en beheren van verlofaanvragen. Dit systeem moet de processen rondom verlofbeheer digitaliseren, automatiseren en verbeteren, zodat medewerkers en managers eenvoudig kunnen communiceren over verlofaanvragen en -statussen.

Scope: De scope van het project omvat de ontwikkeling van een web- en mobiele applicatie die voldoet aan de wensen en eisen van zowel werknemers als managers binnen GeoProfs. Het systeem zal onder andere de volgende functionaliteiten bevatten:

1. **Digitalisering van Verlofaanvragen:**
   * Medewerkers kunnen verlofaanvragen indienen via een online portaal, toegankelijk vanaf computer en smartphone.
2. **Verlofbeheer en Overzicht:**
   * Realtime calculatie van verlofsaldi en overzicht van afwezigheden per afdeling of project.
   * Managers kunnen aanvragen goedkeuren of afkeuren en hebben toegang tot een totaaloverzicht van de verlofplanning binnen hun afdeling.
3. **Automatisering:**
   * Automatische overdracht van afwezigheidsgegevens naar betalingsinstanties.
   * Personalisatie van goedkeuringsprocedures per afdeling of team.
4. **Communicatie:**
   * Meldingen en updates voor zowel medewerkers als managers over de status van verlofaanvragen.
5. **Verlofplanning en -beheer:**
   * Medewerkers kunnen hun verlofrechten inzien en de verlofplanning binnen hun afdeling volgen.
   * Managers kunnen het afwezigheidsniveau in kaart brengen en hun planning daarop aanpassen.

## Achtergrondinformatie over het Project

GeoProfs, opgericht door Johan de Lange, heeft zich in 15 jaar tijd gevestigd als een betrouwbaar en innovatief landmeetkundig bureau. Met een groeiend team van tachtig medewerkers, waaronder specialisten op het gebied van landmeetkunde, Geo ICT, drones en digital image processing, speelt GeoProfs in op de toenemende digitalisering van het vakgebied. De behoefte aan een geavanceerd registratiesysteem voor verlofaanvragen is ontstaan uit de groei van het personeel en de behoefte aan een efficiënter beheer van verlof.

Het project richt zich op het verbeteren van de interne processen door het ontwikkelen van een systeem dat niet alleen de administratieve lasten vermindert, maar ook de communicatie tussen werknemers en management stroomlijnt. Dit draagt bij aan een verhoogde efficiëntie en tevredenheid onder het personeel, terwijl het bedrijf zich kan blijven concentreren op haar kernactiviteiten in de landmeetkunde en aanverwante diensten.

# 2. Systeemarchitectuur

Architectuurdiagram voor GeoProfs:

Dit diagram toont de structuur van het systeem met Next.js als frontend, NestJS als backend, en de database als opslag. De frontend communiceert via API-aanroepen met de backend.

A diagram of a computer program

Description automatically generated

# 3. Technieken en technologieën

We kiezen voor **TypeScript** als programmeertaal, omdat deze helpt bij het opsporen van fouten en de code leesbaarder maakt. Voor de frontend gebruiken we **Next.js**, een framework dat zorgt voor snellere laadtijden door server-side rendering en statische sitegeneratie. Aan de backend kant werken we met **NestJS**, dat op TypeScript is gebaseerd en ons helpt om schaalbare servertoepassingen te bouwen.

Om eenvoudig met de database te werken, gebruiken we **TypeORM**, een tool die de interactie met de database vereenvoudigt. We kiezen voor **MySQL** als database, omdat deze betrouwbaar en snel is voor het opslaan van gestructureerde gegevens.

We volgen de **Scrum**-ontwikkelmethode, die ons in staat stelt om in korte sprints te werken. Dit bevordert samenwerking en maakt het makkelijker om aanpassingen door te voeren.

Het systeem draait volledig in de browser, waardoor het gemakkelijk toegankelijk is voor gebruikers zonder extra software te hoeven installeren. Deze combinatie van technieken zorgt ervoor dat we een robuust en gebruiksvriendelijk registratiesysteem kunnen aanbieden voor het beheer van verlofaanvragen bij GeoProfs.

# 4. Opslag van data

Voor het registratiesysteem van GeoProfs is het belangrijk om een betrouwbare en efficiënte manier van dataopslag te hebben. We kiezen ervoor om alle gegevens op te slaan in een **relationele database**, specifiek **MySQL**.

**Relationele Database (MySQL)**

* **Beschrijving:** MySQL is een populaire relationele database die gestructureerde gegevens opslaat in tabellen. Dit maakt het eenvoudig om gegevens te organiseren en beheren, en het ondersteunt complexe relaties tussen verschillende datatabellen.
* **Voordelen:**
  + **Gegevensintegriteit:** MySQL zorgt voor sterke consistentie en integriteit van gegevens door middel van transacties en referentiële integriteit. Dit is cruciaal voor het beheer van verlofaanvragen, zodat we altijd betrouwbare informatie hebben.
  + **Flexibiliteit:** Het datamodel kan gemakkelijk worden aangepast aan nieuwe vereisten zonder grote impact op de bestaande gegevens. Dit is handig bij veranderende wensen van de gebruikers of bij uitbreiding van functionaliteiten.
  + **Krachtige querymogelijkheden:** Met SQL kunnen we complexe queries uitvoeren om gegevens efficiënt op te halen, zoals het berekenen van verlofsaldi of het genereren van rapporten over afwezigheden.
  + **Schaalbaarheid:** MySQL kan eenvoudig opschalen om grotere hoeveelheden gegevens te verwerken naarmate het bedrijf groeit.

## ERD

**1.** **Entiteiten**

Het model bevat verschillende entiteiten die de basis vormen van het systeem:

* **Admin**:
  + Beheerders van het systeem.
  + Attributen: ID, Naam, Avatar, Email, Wachtwoord.
* **Managers**:
  + Managers hebben de verantwoordelijkheid voor hun eigen afdeling.
  + Attributen: ID, Naam, Avatar, Email, Wachtwoord, Afdeling.
* **Werknemers**:
  + Medewerkers die verlof aanvragen of ziektemeldingen indienen.
  + Attributen: ID, Naam, Avatar, Email, Wachtwoord, Afdeling, Saldo.
* **Afdeling**:
  + Afdelingen waarin werknemers zijn ingedeeld.
  + Attributen: ID, Naam.
* **Groep**:
  + Groeperingen van werknemers voor verschillende doeleinden.
  + Attributen: ID, Naam.
* **Verlofaanvraag**:
  + Bevat alle gegevens met betrekking tot verlofaanvragen die werknemers indienen.
  + Attributen: ID, Begindatum, Einddatum, Type (vakantie, persoonlijk, ziek), Status (goedgekeurd, afgekeurd, in behandeling), Werknemer\_id (verwijst naar de aanvrager).
* **Ziekmelding**:
  + Bevat informatie over ziekmeldingen van werknemers.
  + Attributen: ID, Werknemer\_id, Startdatum, Einddatum.
* **Notificaties**:
  + Meldingen die naar gebruikers worden gestuurd over de status van hun aanvragen of andere belangrijke gebeurtenissen.
  + Attributen: ID, Read (boolean of de notificatie is gelezen), Recipient.

**2. Relaties tussen entiteiten**

Het diagram toont verschillende relaties tussen entiteiten:

* **Managers en Werknemers**:
  + Werknemers hebben een afdeling toegewezen via een veel-op-een-relatie met de entiteit Afdeling.
  + Managers hebben een een-op-een relatie met de entiteit Afdeling.
* **Managers en Werknemers**:
  + Managers hebben een relatie met werknemers binnen hun afdeling. Dit wordt geïmpliceerd door de aanwezigheid van de afdeling in beide entiteiten.
* **Verlofaanvraag**:
  + Elke verlofaanvraag is gekoppeld aan één werknemer via het veld Werknemer\_id. Werknemers kunnen meerdere verlofaanvragen hebben, wat duidt op een een-op-veel-relatie.
* **Ziekmelding**:
  + Elke ziekmelding is ook gekoppeld aan één werknemer via Werknemer\_id.
* **Notificaties**:
  + Notificaties kunnen worden gestuurd naar managers, werknemers of mogelijk andere entiteiten. De relatie met de recipient is geïmpliceerd door het veld Recipient.

**3. Beschrijving van bedrijfsprocessen**

Met dit datamodel kan het systeem de volgende processen ondersteunen:

* **Afdelingsbeheer**: Admins kunnen afdelingen aanmaken en beheren. Werknemers en managers worden gekoppeld aan een afdeling.
* **Verlofbeheer**: Werknemers kunnen verlofaanvragen indienen, die vervolgens door managers of het systeem kunnen worden goedgekeurd of afgewezen. De status van de aanvragen wordt beheerd, en gebruikers ontvangen notificaties over wijzigingen in de status.
* **Ziektebeheer**: Werknemers kunnen ziekmeldingen indienen, en deze meldingen worden bijgehouden met start- en einddata. Dit kan verder worden uitgebreid met bijvoorbeeld herstelmeldingen of doktersverklaringen.
* **Notificatiebeheer**: Het systeem verstuurt notificaties naar gebruikers (werknemers of managers), bijvoorbeeld als een verlofaanvraag is goedgekeurd of afgewezen. Notificaties worden gemarkeerd als gelezen of ongelezen.

A diagram of a company

Description automatically generated

# 5. Toegankelijkheid en bereikbaarheid van gegevens

De toegang tot gegevens is zorgvuldig ingericht en gereguleerd. Intern hebben zowel managers als beheerders specifieke toegang tot gegevens binnen het systeem. Deze toegang stelt hen in staat om belangrijke inzichten te verkrijgen en strategische beslissingen te nemen op basis van actuele informatie. Voor de externe toegang tot gegevens hebben we momenteel geen integraties met externe systemen. Dit betekent dat alle gegevensbeheer en -analyse intern plaatsvinden, wat de veiligheid en integriteit van onze gegevens waarborgt. Toegangsbeheer Om de toegang tot gegevens effectief te beheren, maken we gebruik van een rollen- en rechten systeem. Dit systeem definieert verschillende gebruikersrollen, waaronder managers, gebruikers en beheerders. Elke rol heeft specifieke rechten en autorisaties, wat ervoor zorgt dat alleen bevoegde personen toegang hebben tot bepaalde gegevens en functionaliteiten. Deze aanpak helpt ons niet alleen om de gegevens te beschermen, maar ook om ervoor te zorgen dat gebruikers alleen toegang hebben tot de informatie die zij nodig hebben voor hun werkzaamheden. Koppelingen met Andere Systemen Binnen de huidige scope van onze gegevensverwerking zijn er geen koppelingen met andere systemen. Dit biedt ons de mogelijkheid om ons volledig te concentreren op interne processen en systemen zonder externe invloeden. In de toekomst kunnen er mogelijkheden worden onderzocht om integraties of API-koppelingen te implementeren, maar op dit moment ligt de focus op de interne gegevensverwerking en -toegang.6. Beveiliging

Het authenticatie- en autorisatiemechanisme wordt opgebouwd rond een rollen-gebaseerd toegangscontrolesysteem (RBAC) en een standaard inlogsysteem met gebruikersnaam en wachtwoord.

Authenticatieproces:

* Gebruikersnaam & Wachtwoord: Bij het inloggen voert de gebruiker een combinatie van gebruikersnaam en wachtwoord in. Het wachtwoord wordt nooit in platte tekst opgeslagen, maar wordt gehasht met een veilige hashfunctie zoals bcrypt, waarbij ook een salt wordt toegevoegd.
* Rate Limiting: Om brute force-aanvallen te mitigeren, wordt rate limiting toegepast. Dit betekent dat het aantal mislukte inlogpogingen per gebruiker binnen een bepaald tijdsinterval wordt beperkt. Bij overschrijding van dit limiet wordt het account tijdelijk geblokkeerd of wordt de wachttijd voor nieuwe inlogpogingen verhoogd.

Autorisatie:

* Rollen-gebaseerd systeem: Het autorisatiesysteem maakt gebruik van rollen zoals admin, manager, en werknemer. Elke rol heeft specifieke rechten en toegangsniveaus. Deze rollen worden centraal beheerd in het systeem.
  + Admins hebben volledige toegang tot alle functies.
  + Managers hebben beperkte rechten.
  + Werknemers hebben zeer beperkte toegang.

## Data encryptie

Om de integriteit van gebruikersgegevens te waarborgen, wordt encryptie toegepast voor data in rust.

### Data in rust

Gevoelige gegevens zoals wachtwoorden zullen worden opgeslagen als hash met bcrypt. Dit maakt het onmogelijk om de oorspronkelijke gegevens eenvoudig terug te halen.

### Data in transport

De applicatie is op dit moment in development. Er is daarom gekozen om nu nog geen HTTPS-verbinding op te zetten tussen de verschillende componenten. Echter, wanneer de applicatie naar een productieomgeving wordt overgebracht, zal TLS (Transport Layer Security) gebruikt worden om alle netwerkcommunicatie te versleutelen, wat ervoor zorgt dat gegevens niet onderschept en gemanipuleerd kunnen worden tijdens transport.

## Logging

Logging is essentieel voor het waarborgen van de veiligheid en stabiliteit van een applicatie. Doormiddel van logging worden belangrijke gebeurtenissen binnen het systeem vastgelegd. Dit helpt bij het identificeren van fouten en verdachte activiteiten. Logging vormt daarmee een cruciaal onderdeel van de applicatiebeveiliging en het fout opsporingsproces.

# 7. Prestaties en schaalbaarheid

Verwerkingssnelheid  
De prestaties van het verlofregistratiesysteem zijn cruciaal voor een gebruiksvriendelijke ervaring. Enkele prestatiedoelstellingen zijn:

* Laadtijden: De frontend van de applicatie (Next.js) moet in staat zijn om binnen 2 seconden te laden, ongeacht het apparaat dat de gebruiker gebruikt. Dit wordt bereikt door server-side rendering en het optimaliseren van statische bestanden.
* Responstijden: API-aanroepen naar de backend (NestJS) moeten binnen 200 milliseconden worden afgehandeld. Dit is essentieel voor interacties zoals het indienen van verlofaanvragen en het ophalen van verlofsaldi.

# 8. Onderhoudbaarheid

Documentatie: Bijhouden van duidelijke technische documentatie en richtlijnen, zodat het ontwikkelteam efficiënt kan samenwerken en nieuwe leden snel kunnen inwerken.

Om storingen te minimaliseren en prestaties te verbeteren:

**Code Reviews**: Regelmatige beoordelingen van de code om best practices te waarborgen en potentiële problemen vroegtijdig te identificeren.

**Testautomatisering**: Implementatie van geautomatiseerde tests om de werking van nieuwe en bestaande functionaliteiten te garanderen.

**Componentintegratie**: Regelmatig testen van de integratie tussen verschillende systeemcomponenten om ervoor te zorgen dat ze correct samenwerken en om integratiefouten vroegtijdig op te sporen.

**End-to-End (e2e) Tests**: Uitvoeren van e2e-tests om de volledige gebruikerservaring te valideren, van het indienen van een verlofaanvraag tot de uiteindelijke goedkeuring, en om ervoor te zorgen dat alle systeemonderdelen naar behoren functioneren.

Met deze aanpak waarborgt GeoProfs de onderhoudbaarheid en effectiviteit van het verlofregistratiesysteem.

# **9.** Koppelingen met andere systemen

In de huidige scope van het verlofregistratiesysteem van GeoProfs is er geen noodzaak voor koppelingen met externe systemen. Echter, er moet wel een mogelijkheid zijn voor toekomstige integraties in de volgende fase van het project. Daarnaast is het belangrijk dat managers de gegevens eenvoudig kunnen downloaden in een bestand (bijvoorbeeld CSV of Excel), zodat ze deze kunnen gebruiken voor rapportage of verdere verwerking. Deze functionaliteit biedt flexibiliteit en bereidheid voor toekomstige uitbreidingen zonder de huidige processen te verstoren.

# 10. Klassendiagram

# 11. Documentatie

**Gebruikersdocumentatie**: Handleidingen voor eindgebruikers.

**Technische documentatie**: Gedetailleerde documentatie voor ontwikkelaars, inclusief API-documentatie en installatierichtlijnen.