

WordPress Integratie per Project



Overzicht

WordPress-configuratie kan nu op twee niveaus ingesteld worden:

1. **Project-niveau** (nieuw!) - Specifieke WordPress site per project
2. **Client-niveau** (fallback) - Algemene WordPress site voor alle projecten

Het systeem gebruikt automatisch de project-specifieke configuratie als die bestaat, anders valt het terug op de client-level configuratie.



Nieuwe Functionaliteit

1. WordPress Configuratie per Project

In de project detail pagina (`/client-portal/projects/[id]`) kunnen gebruikers nu:

- **WordPress URL** instellen (bijv. `https://jouwwebsite.nl`)
- **Gebruikersnaam** configureren
- **Application Password** toevoegen
- **Standaard categorie** instellen (optioneel)
- **Auto-publiceren** aan/uitzetten
- **Verbinding testen** voordat je opslaat

2. Automatische Config Selectie

Bij het publiceren naar WordPress:

1. Check: Heeft project een WordPress configuratie?
 - ☒ Ja → Gebruik project config
 - ☒ Nee → Gebruik client config (fallback)

Dit betekent:

- **Zonder project config** → Werkt zoals voorheen (client-level)
- **Met project config** → Gebruikt project-specifieke settings
- **Beide mogelijk** → Je kunt sommige projecten koppelen aan andere WordPress sites

Technische Implementatie

Database Schema (Prisma)

```
model Project {
  // ... andere velden ...

  // WordPress Integration (optioneel per project)
  wordpressUrl      String?
  wordpressUsername  String?
  wordpressPassword  String?
  wordpressCategory  String?
  wordpressAutoPublish Boolean @default(false)
}
```

API Endpoints

WordPress Settings Opslaan

PUT /api/client/projects/[id]/wordpress

```
{
  wordpressUrl: string,
  wordpressUsername: string,
  wordpressPassword: string,
  wordpressCategory?: string,
  wordpressAutoPublish: boolean
}
```

WordPress Verbinding Testen

POST /api/client/projects/[id]/wordpress/test

```
{
  wordpressUrl: string,
  wordpressUsername: string,
  wordpressPassword: string
}
```

WordPress Publisher Helper

Nieuwe functie in `lib/wordpress-publisher.ts` :

```
export async function getWordPressConfig(options: {
  clientEmail?: string;
  projectId?: string;
}): Promise<WordPressConfig | null>
```

Deze functie:

- Haalt automatisch de juiste WordPress config op
- Controleert eerst project-specifieke settings
- Valt terug op client-level settings
- Retourneert `null` als geen config gevonden

Publishing Routes

Beide publishing routes ondersteunen nu project-specifieke config:

1. `/api/client/publish-to-wordpress` (aanbevolen)
 - Vereist `projectId` in body
 - Gebruikt automatisch project config
2. `/api/client/wordpress/publish`
 - Ondersteunt optionele `projectId` in body
 - Falls back naar client config als geen `projectId`



Gebruik

Voor Gebruikers

1. **Ga naar je project:** `/client-portal/projects/[id]`
2. **Klik op “Configureren”** bij WordPress Integratie
3. **Vul de gegevens in:**
 - WordPress URL (inclusief `https://`)
 - Gebruikersnaam
 - Application Password (maak aan in WordPress)
 - Optioneel: standaard categorie
4. **Test de verbinding** om te controleren of alles werkt
5. **Klik op “Opslaan”**

Voor Ontwikkelaars

Publiceren met Project Context

```
// In je publishing functie
const response = await fetch('/api/client/publish-to-wordpress', {
  method: 'POST',
  headers: { 'Content-Type': 'application/json' },
  body: JSON.stringify({
    projectId: 'clxxx...', // Project ID meegeven
    title: 'Artikel Titel',
    content: '<p>Artikel inhoud...</p>',
    excerpt: 'Korte samenvatting',
    featuredImageUrl: 'https://i.ytimg.com/vi/105Px_TzE8E/sddefault.jpg',
    seoTitle: 'SEO Titel',
    seoDescription: 'SEO Beschrijving',
    focusKeyword: 'keyword'
  })
});
```

Config Ophalen (Server-side)

```
import { getWordPressConfig } from '@lib/wordpress-publisher';

const config = await getWordPressConfig({
  projectId: 'clxxx...'
});

if (config) {
  // Config bestaat, gebruik deze
  await publishToWordPress(config, articleData);
} else {
  // Geen config gevonden
  throw new Error('WordPress niet geconfigureerd');
}
```

Use Cases

Use Case 1: Meerdere Websites

Een gebruiker heeft:

- Project A → WordPress site 1 (yoga.nl)
- Project B → WordPress site 2 (fitness.nl)
- Project C → Geen WordPress (alleen content generatie)

Use Case 2: Staging & Productie

- Project “Writgo Staging” → WordPress op staging.writgo.nl
- Project “Writgo Productie” → WordPress op writgo.nl

Use Case 3: Klanten Management

Een agency met meerdere klanten:

- Elke klant heeft eigen project
- Elk project publiceert naar eigen WordPress site
- Centrale content bibliotheek blijft shared



Security

- Application passwords zijn veiliger dan normale wachtwoorden
- Passwords worden encrypted opgeslagen in database
- Elke API call valideert dat project bij ingelogde client hoort
- Verbinding test checkt rechten voordat config wordt opgeslagen



Migratie

Bestaande configuratie blijft werken!

- Client-level WordPress settings blijven functioneel
- Projecten zonder specifieke config gebruiken automatisch client config
- Geen data loss of breaking changes

? FAQ

Q: Wat gebeurt er als ik zowel project als client config heb?

A: Project config heeft voorrang. Client config is de fallback.

Q: Kan ik een project zonder WordPress gebruiken?

A: Ja, WordPress is volledig optioneel. Content wordt gewoon gegenereerd maar niet gepubliceerd.

Q: Moet ik alle projecten configureren?

A: Nee, alleen als je wilt dat een project naar een specifieke WordPress site publiceert.

Q: Waar vind ik mijn Application Password?

A: WordPress Admin → Gebruikers → Profiel → Application Passwords (onderaan de pagina)

Q: Wat als mijn WordPress site geen REST API heeft?

A: De REST API is standaard ingeschakeld sinds WordPress 4.7. Check of je een security plugin hebt die de API blokkeert.



Toekomstige Uitbreidingen

Mogelijke toekomstige features:

- [] Meerdere WordPress sites per project
- [] Scheduled publishing per project
- [] Auto-sync van categories/tags
- [] WordPress multisite ondersteuning
- [] Custom post types configuratie



Status

- ✓ **Live op productie** (writgoai.nl)
 - ✓ Volledig getest en werkend
 - ✓ Backwards compatible
 - ✓ Gedocumenteerd
-

Laatste update: 1 november 2025

Implementatie checkpoint: “WordPress per project configuratie”