

Email Management System - Fase 1 Documentatie

Overzicht

Dit document beschrijft de implementatie van **Fase 1** van het Email Management Systeem voor Writago. Deze fase legt de basis voor email beheer met IMAP integratie, email configuratie, inbox weergave en email detail views.

Architectuur

Database Schema

Het systeem gebruikt de volgende tabellen (gedefinieerd in `supabase/migrations/20251210_email_inbox_tables.sql`):

MailboxConnection - Email Account Configuratie

- id (`TEXT`, **PRIMARY KEY**)
- clientId (`TEXT`, **FOREIGN KEY** ↗ Client)
- provider (`TEXT`) - 'imap', 'gmail', 'outlook'
- email (`TEXT`) - Email adres
- displayName (`TEXT`) - Weergave naam
- imapHost, imapPort, imapTls - IMAP settings
- smtpHost, smtpPort, smtpTls - SMTP settings
- password (`TEXT`) - **Encrypted** password
- accessToken, refreshToken, tokenExpiry - OAuth tokens (voor latere fases)
- isActive (`BOOLEAN`)
- lastSyncAt (`TIMESTAMP`)
- lastError (`TEXT`)

InboxEmail - Email Messages (voor toekomstige fases)

- Opslag van fetched emails
- AI analysis velden
- Invoice detection
- Thread management

Beveiliging

Encryption

- **Library:** Native Node.js `crypto` module
- **Algorithm:** AES-256-GCM
- **Key Storage:** Environment variable `EMAIL_ENCRYPTION_KEY`
- **Format:** `iv:authTag:encrypted` (alle in hex)

Implementatie in `nextjs_space/lib/encryption.ts`:

```
export function encrypt(text: string): string
export function decrypt(encryptedData: string): string
export function generateEncryptionKey(): string // Voor productie key generatie
```

BELANGRIJK: Genereer een nieuwe encryption key voor productie:

```
node -e "console.log(require('crypto').randomBytes(32).toString('hex'))"
```

Voeg dit toe aan je environment variables als `EMAIL_ENCRYPTION_KEY`.

Componenten

1. IMAP Client (`lib/email/imap-client.ts`)

Verantwoordelijk voor alle IMAP operaties.

Dependencies

- `imapflow` - Modern IMAP client met `async/await`
- `mailparser` - Email parsing (headers, body, attachments)

Functies

```
// Test IMAP connectie
testConnection(config: IMAPConfig): Promise<{ success: boolean; message: string; error?: string }>

// Fetch emails van inbox
fetchEmails(config: IMAPConfig, options?: FetchEmailsOptions): Promise<{
  success: boolean;
  emails?: EmailMessage[];
  error?: string;
  total?: number
}>

// Fetch single email by UID
fetchEmailById(config: IMAPConfig, uid: number, folder?: string): Promise<{
  success: boolean;
  email?: EmailMessage;
  error?: string
}>

// Mark email als gelezen/ongelezen
markAsRead(config: IMAPConfig, uid: number, folder?: string): Promise<{ success: boolean; error?: string }>
markAsUnread(config: IMAPConfig, uid: number, folder?: string): Promise<{ success: boolean; error?: string }>

// List folders/mailboxes
listFolders(config: IMAPConfig): Promise<{ success: boolean; folders?: string[]; error?: string }>
```

Features

- Automatische reconnectie handling
- Graceful error handling

- HTML en plain text body parsing
 - Attachment metadata extraction
 - Read/unread status
-

2. API Routes

POST /api/admin/email/accounts

Aanmaken of updaten van email account.

Body:

```
{
  "id": "optional-for-update",
  "email": "info@writgo.nl",
  "displayName": "Writgo Support",
  "imapHost": "imap.gmail.com",
  "imapPort": 993,
  "imapTls": true,
  "smtpHost": "smtp.gmail.com",
  "smtpPort": 587,
  "smtpTls": true,
  "password": "your-app-password",
  "provider": "imap",
  "isActive": true
}
```

Response:

```
{
  "success": true,
  "account": { /* account object with masked password */ }
}
```

GET /api/admin/email/accounts

Ophalen van alle email accounts.

Query Params:

- clientId (optional) - Filter op client

Response:

```
{
  "accounts": [
    { /* account objects with masked passwords */ }
  ]
}
```

POST /api/admin/email/test-connection

Test IMAP connectie.

Body:

```
{
  "imapHost": "imap.gmail.com",
  "imapPort": 993,
  "username": "info@writgo.nl",
  "password": "your-app-password",
  "tls": true
}
```

Response:

```
{
  "success": true,
  "message": "Successfully connected to imap.gmail.com. Found 5 mailboxes."
}
```

GET /api/admin/email/inbox

Ophalen van emails van inbox via IMAP.

Query Params:

- accountId (optional) - Specifiek account, anders eerste active account
- folder (optional, default: 'INBOX')
- limit (optional, default: 50)
- unreadOnly (optional, default: false)

Response:

```
{
  "success": true,
  "emails": [ /* array of email objects */ ],
  "total": 150,
  "account": {
    "id": "...",
    "email": "info@writgo.nl",
    "displayName": "Writgo Support"
  }
}
```

GET /api/admin/email/message

Ophalen van single email by UID.

Query Params:

- uid (required) - Email UID
- accountId (optional)
- folder (optional, default: 'INBOX')
- markRead (optional, default: false) - Auto-mark as read

Response:

```
{
  "success": true,
  "email": { /* full email object with body */ }
}
```

POST /api/admin/email/message

Mark email als read/unread.

Body:

```
{
  "accountId": "optional",
  "uid": 12345,
  "folder": "INBOX",
  "action": "markRead" // or "markUnread"
}
```

3. Admin Interface

Navigatie

Email sectie toegevoegd aan `lib/admin-navigation-config.ts`:

```
{
  title: 'Email',
  items: [
    {
      label: 'Inbox',
      href: '/admin/email/inbox',
      icon: Inbox,
      description: 'Email inbox',
    },
    {
      label: 'Instellingen',
      href: '/admin/email/instellingen',
      icon: Settings,
      description: 'Email configuratie',
    },
  ],
}
```

Email Instellingen (</admin/email/instellingen>)

Features:

- Email account configuratie formulier
- IMAP/SMTP settings
- Test connectie functionaliteit
- Opslaan met password encryption
- Pre-fill existing configuration
- Help card met common providers (Gmail, Outlook)

UI Components:

- Email adres & display name
- IMAP settings (host, port, TLS)
- SMTP settings (host, port, TLS)
- Password field (encrypted bij opslaan)
- Test Connectie button
- Opslaan button
- Help sectie

Email Inbox (/admin/email/inbox)

Features:

- Lijst van emails (laatste 50)
- Email preview (van, onderwerp, snippet, datum)
- Read/unread status indicator
- Attachment indicator
- Refresh button (fetch latest)
- Click om email te openen
- Link naar settings als geen account

UI Components:

- Header met account info
- Ververs button
- Email tabel met:
 - Icon (gelezen/ongelezen)
 - Van (naam + email)
 - Onderwerp
 - Snippet (eerste 200 chars)
 - Datum (relatief: vandaag = tijd, anders datum)
 - Badges (ongelezen, bijlage)

Email Detail (/admin/email/inbox/[uid])

Features:

- Volledige email weergave
- Headers (van, aan, cc, datum)
- HTML/Text body toggle
- Attachments lijst
- Auto-mark as read
- Terug naar inbox button

UI Components:

- Back button
- Email header card (van, aan, cc, datum)
- Attachments card (als aanwezig)
- Email body card (HTML/text toggle)

Setup & Configuratie

1. Environment Variables

Voeg toe aan .env.local :

```
# Email Encryption Key (generate met: node -e "console.log(require('crypto').randomBytes(32).toString('hex'))")
EMAIL_ENCRYPTION_KEY=your-64-character-hex-key-here
```

2. Dependencies

Dependencies zijn al geïnstalleerd:

```
{
  "imapflow": "^1.0.156",
  "mailparser": "^3.7.1"
}
```

3. Database Migration

De database migration is al aanwezig in `supabase/migrations/20251210_email_inbox_tables.sql`.

Om toe te passen (als nog niet gedaan):

```
# Via Supabase CLI
supabase db push

# Of via SQL in Supabase Dashboard
# Kopieer inhoud van migration file en voer uit
```

4. Gmail Setup (Voorbeeld)

Voor Gmail moet je een **App-specifiek wachtwoord** gebruiken:

1. Ga naar Google Account → Security
2. Enable 2-Step Verification (vereist)
3. Ga naar “App passwords”
4. Genereer nieuw wachtwoord voor “Mail”
5. Gebruik dit wachtwoord in de configuratie

Settings:

- IMAP Host: `imap.gmail.com`
- IMAP Port: `993`
- SMTP Host: `smtp.gmail.com`
- SMTP Port: `587`
- TLS: Enabled

Gebruik Flow

Eerste Keer Setup

1. **Ga naar Email Instellingen** (`/admin/email/instellingen`)
2. **Vul email configuratie in:**
 - Email adres
 - IMAP host en port
 - Password (voor Gmail: app-specifiek wachtwoord)
3. **Test connectie** (optioneel maar aanbevolen)
4. **Opslaan**

Email Lezen

1. **Ga naar Email Inbox** (`/admin/email/inbox`)
2. **Emails worden automatisch geladen** via IMAP
3. **Klik op een email** om te openen

4. **Email wordt geopend** in detail view
5. **Email wordt automatisch gemarkerd als gelezen**

Troubleshooting

“No email account configured”

→ Ga naar Email Instellingen en configureren account

“Failed to decrypt password”

→ EMAIL_ENCRYPTION_KEY is gewijzigd of niet ingesteld
→ Configureer email account

“Failed to connect to IMAP server”

→ Check IMAP host, port, username, password
→ Voor Gmail: gebruik app-specifiek wachtwoord
→ Check firewall/network settings

Testing

Manual Testing Checklist

- [] Email account aanmaken in settings
- [] Test connectie succesvol
- [] Account opslaan
- [] Inbox openen
- [] Emails worden geladen
- [] Email openen
- [] Email body wordt getoond
- [] Attachments worden getoond (als aanwezig)
- [] Terug naar inbox
- [] Refresh emails

Security Testing

- [] Password wordt encrypted opgeslagen in database
 - [] Password wordt niet teruggestuurd naar frontend (masked)
 - [] Decryption werkt correct
 - [] Environment variable check werkt
-

Bekende Beperkingen (Fase 1)

1. **Geen email versturen** - SMTP is geconfigureerd maar niet geïmplementeerd (Fase 2)
2. **Geen email persistence** - Emails worden real-time van IMAP gehaald, niet opgeslagen in database (Fase 2)
3. **Geen AI analysis** - Database velden zijn er, maar functionaliteit komt in Fase 2
4. **Geen attachment downloads** - Alleen metadata wordt getoond (Fase 2)
5. **Geen thread grouping** - Emails worden als individuele items getoond (Fase 2)

6. **Geen search/filter** - Alleen basis inbox view (Fase 2)
 7. **Single account** - Alleen eerste active account wordt gebruikt (multi-account in Fase 2)
-

Volgende Fases (Roadmap)

Fase 2: Email Persistence & AI Analysis

- Emails opslaan in database (InboxEmail tabel)
- AI analysis (categorie, prioriteit, sentiment, suggested reply)
- Invoice detection
- Background sync job
- Search & filtering

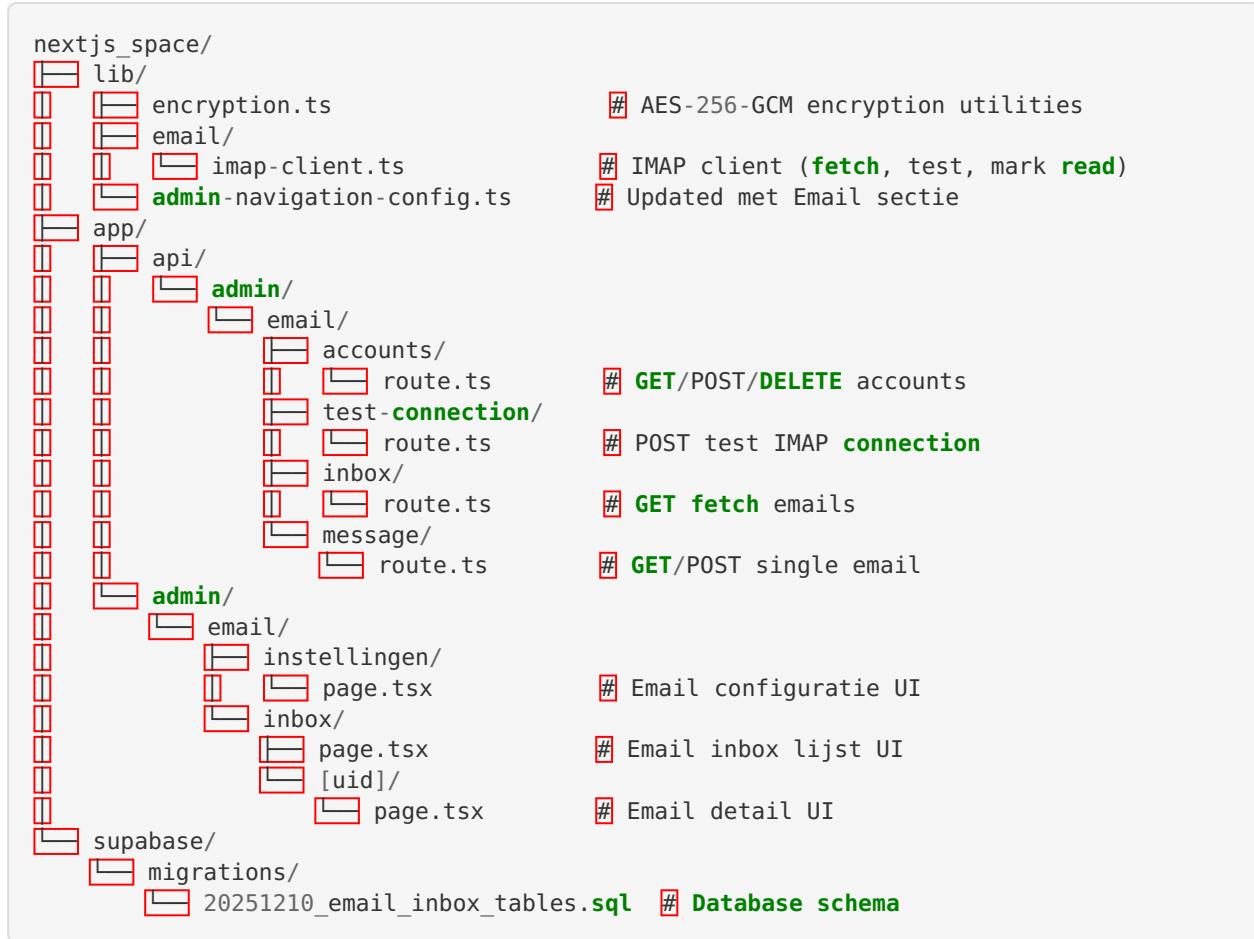
Fase 3: Email Versturen & Replies

- SMTP client implementatie
- Email composer UI
- Reply functionaliteit
- Forward functionaliteit
- Templates

Fase 4: Advanced Features

- Thread grouping
 - Auto-reply configuratie
 - Multiple accounts
 - Email rules/automation
 - Attachment downloads
-

Code Structuur



Onderhoud & Monitoring

Logs

- API routes loggen errors naar console
- IMAP errors worden gevangen en gereturned als `{ success: false, error: "..." }`

Database

- `MailboxConnection.lastSyncAt` - Laatste sync tijdstip
- `MailboxConnection.lastError` - Laatste error (null bij success)

Performance

- IMAP requests zijn real-time (niet gecached in Fase 1)
- Limit tot 50 emails per request
- Timeout: IMAP default (30s)

Support & Contact

Voor vragen of problemen met het email systeem:

- Check deze documentatie
 - Check bekende beperkingen
 - Check troubleshooting sectie
 - Test met `Test Connectie` button in settings
-

Changelog

v1.0.0 (Fase 1) - December 11, 2025

- Database schema (MailboxConnection, InboxEmail)
 - Encryption utilities (AES-256-GCM)
 - IMAP client (imapflow + mailparser)
 - API routes (accounts, test, inbox, message)
 - Admin UI (instellingen, inbox, detail)
 - Navigation update
 - Documentation
-

Einde van Fase 1 Documentatie