Email Notification System Documentation

Overview

The Mindful Champion platform now includes a comprehensive automated email notification system that sends beautiful, professional emails to users when their video analysis is complete. The system includes full admin controls for monitoring, managing, and troubleshooting email delivery.

Features

🎯 Core Features

1. **Automated Email Sending**
   - Automatically sends emails when video analysis completes
   - Beautiful, responsive HTML email templates
   - Includes analysis highlights and key metrics
   - Direct links to view full analysis

2. **Email Tracking**
   - Track email status (pending, sent, delivered, opened, clicked, failed)
   - Monitor delivery rates and open rates
   - Log all email attempts and errors
   - Track retry attempts for failed emails

3. **Admin Dashboard**
   - Comprehensive email management interface at `/admin/email-notifications`
   - Real-time statistics and analytics
   - Filter and search capabilities
   - Export to CSV functionality
   - Resend failed emails with one click

4. **Email Settings Control**
   - Enable/disable email notifications globally
   - Toggle specific email types (video analysis, welcome, etc.)
   - Configure sender information
   - Set retry limits and delays
   - Manage from email and reply-to addresses

Architecture

Database Schema

EmailNotification Model

```
model EmailNotification {
  id                String              @id @default(cuid())
  userId            String
  videoAnalysisId   String?
  type              EmailNotificationType
  recipientEmail    String
  recipientName     String?
  subject           String
  htmlContent       String
  textContent       String?
  status            EmailStatus         @default(PENDING)
  sentAt            DateTime?
  deliveredAt       DateTime?
  openedAt          DateTime?
  clickedAt         DateTime?
  failedAt          DateTime?
  error             String?
  metadata          Json?
  resendEmailId     String?
  retryCount        Int                 @default(0)
  lastRetryAt       DateTime?
  createdAt         DateTime            @default(now())
  updatedAt         DateTime            @updatedAt
  user              User
  videoAnalysis     VideoAnalysis?
}
```
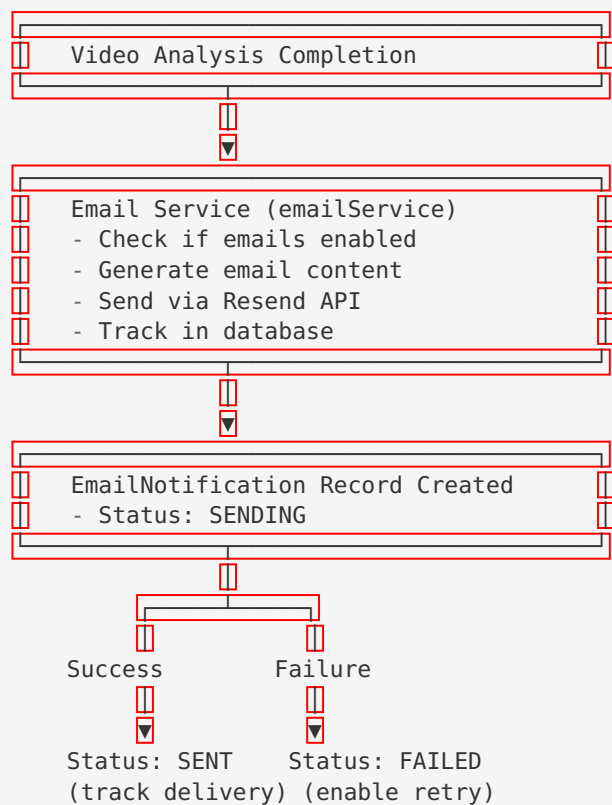
VideoAnalysis Updates

```
model VideoAnalysis {
  // ... existing fields ...
  emailNotificationSent       Boolean           @default(false)
  emailNotificationSentAt     DateTime?
  emailNotificationStatus     EmailStatus?
  emailNotificationError      String?
  emailNotifications          EmailNotification[]
}
```

EmailSettings Model

```
model EmailSettings {
  id                         String   @id @default(cuid())
  emailNotificationsEnabled  Boolean  @default(true)
  videoAnalysisEmailsEnabled Boolean  @default(true)
  welcomeEmailsEnabled       Boolean  @default(true)
  marketingEmailsEnabled     Boolean  @default(false)
  maxRetryAttempts           Int      @default(3)
  retryDelayMinutes          Int      @default(30)
  fromEmail                  String   @default("noreply@mindfulchampion.com")
  fromName                   String   @default("Mindful Champion")
  replyToEmail               String?
  createdAt                  DateTime @default(now())
  updatedAt                  DateTime @updatedAt
}
```

Email Service Architecture

```
┌─────────────────────────────────────┐
│       Video Analysis Completion      │
└─────────────────────────────────────┘
                   │
                   ▼
┌─────────────────────────────────────┐
│       Email Service (emailService)   │
│       - Check if emails enabled      │
│       - Generate email content       │
│       - Send via Resend API          │
│       - Track in database            │
└─────────────────────────────────────┘
                   │
                   ▼
┌─────────────────────────────────────┐
│    EmailNotification Record Created  │
│    - Status: SENDING                 │
└─────────────────────────────────────┘
                   │
          ┌────────┴────────┐
          │                 │
       Success           Failure
          │                 │
          ▼                 ▼
    Status: SENT      Status: FAILED
    (track delivery)  (enable retry)
```

Configuration

Environment Variables

Add to your `.env` or `.env.local` file:

```
# Email Service (Resend)
RESEND_API_KEY=your_resend_api_key_here

# Optional: Custom from email
RESEND_FROM_EMAIL=noreply@yourdomain.com
```

Resend Setup

1. **Sign up for Resend**
   - Visit https://resend.com
   - Create an account
   - Get your API key from the dashboard

2. **Verify Domain** (for production)
   - Add your domain in Resend dashboard
   - Add required DNS records
   - Verify domain ownership

3. **Configure API Key**
   - Add `RESEND_API_KEY` to your environment variables
   - For development, the system uses a mock client if no key is configured

Initial Setup

The system will automatically:
- Create default `EmailSettings` on first use
- Generate Prisma client with new models
- Initialize email service with sensible defaults

Usage

For Developers

Sending Custom Emails

```
import { emailService } from '@/lib/email/email-service'

await emailService.sendEmail({
  userId: user.id,
  recipientEmail: user.email,
  recipientName: user.name,
  subject: 'Your Subject Here',
  htmlContent: '<html>...</html>',
  type: 'VIDEO_ANALYSIS_COMPLETE',
  videoAnalysisId: analysis.id,
  metadata: {
    customField: 'value'
  }
})
```

Creating Custom Email Templates

```
import { generateViewAnalysisButton, formatList } from '@/lib/email/email-helpers'

export function generateCustomEmail(data: CustomEmailData): string {
  return `
    <!DOCTYPE html>
    <html>
      <body>
        ${generateViewAnalysisButton(data.analysisId)}
        ${formatList(data.items, 'strengths')}
      </body>
    </html>
  `
}
```

Retry Failed Emails Programmatically

```
import { emailService } from '@/lib/email/email-service'

const result = await emailService.retryEmail(emailNotificationId)

if (result.success) {
  console.log('Email resent successfully')
} else {
  console.error('Retry failed:', result.error)
}
```

For Admins

Accessing the Dashboard

1. Navigate to `/admin` (requires ADMIN role)
2. Click on "Email Notifications" tab or button
3. Or go directly to `/admin/email-notifications`

Managing Email Notifications

View All Notifications

- See list of all emails sent by the system
- View status, recipients, subjects, and timestamps
- Filter by status, type, or date range
- Search by email, name, or subject

Email Statistics

- Total sent, failed, pending counts
- Delivery rate, open rate, click rate
- Failure rate analysis
- Real-time metrics

Resend Failed Emails

- Click "Resend" button on any failed email
- System tracks retry attempts
- Respects max retry limits from settings

Configure Settings

- Toggle email notifications on/off

- Enable/disable specific email types
- Configure sender information
- Set retry limits and delays
- Update from email and name

Export Data
- Click "Export CSV" to download email logs
- Includes all notification data
- Useful for analysis and reporting

Email Templates

Video Analysis Complete Email

Subject: 🎾 Your Video Analysis is Ready!

Content Sections:
1. **Header** - Mindful Champion branding
2. **Hero** - Celebration icon and message
3. **Overall Score** - Large score badge with rating
4. **Key Metrics** - Total shots, duration, analysis type
5. **Top Strengths** (3 items) - Green checkmarks
6. **Priority Focus Areas** (3 items) - Orange warnings
7. **Teaser** - What's included in full analysis
8. **CTA Button** - View Full Analysis (prominent)
9. **Footer** - Links to video library, settings, support

Responsive Design:
- Mobile-optimized layout
- Adapts to different screen sizes
- Maintains branding on all devices

API Endpoints

Admin Endpoints

GET /api/admin/email-notifications

Description: List all email notifications with filters

Query Parameters:
- `page` (number) - Page number for pagination
- `limit` (number) - Items per page (default: 20)
- `status` (EmailStatus) - Filter by status
- `type` (EmailNotificationType) - Filter by type
- `search` (string) - Search email, name, or subject
- `startDate` (ISO date) - Filter from date
- `endDate` (ISO date) - Filter to date

Response:

```json
{
  "notifications": [
    {
      "id": "...",
      "recipientEmail": "user@example.com",
      "subject": "Your Video Analysis is Ready!",
      "status": "SENT",
      "sentAt": "2025-11-08T10:00:00Z",
      "user": { ... },
      "videoAnalysis": { ... }
    }
  ],
  "pagination": {
    "total": 100,
    "page": 1,
    "limit": 20,
    "totalPages": 5
  }
}
```

GET /api/admin/email-notifications/stats

Description: Get email notification statistics

Query Parameters:
- `startDate` (ISO date) - Optional start date filter
- `endDate` (ISO date) - Optional end date filter
- `type` (EmailNotificationType) - Optional type filter

Response:

```json
{
  "stats": {
    "total": 1000,
    "sent": 950,
    "failed": 30,
    "pending": 20,
    "delivered": 900,
    "opened": 450,
    "clicked": 150,
    "deliveryRate": "95.0%",
    "openRate": "47.4%",
    "clickRate": "15.8%",
    "failureRate": "3.0%"
  }
}
```

POST /api/admin/email-notifications/resend

Description: Resend a failed email notification

Body:

```json
{
  "emailId": "notification_id"
}
```

Response:

```json
{
  "success": true,
  "message": "Email resent successfully"
}
```

GET /api/admin/email-notifications/settings

Description: Get email settings

Response:

```json
{
  "settings": {
    "id": "...",
    "emailNotificationsEnabled": true,
    "videoAnalysisEmailsEnabled": true,
    "maxRetryAttempts": 3,
    "fromEmail": "noreply@mindfulchampion.com",
    "fromName": "Mindful Champion"
  }
}
```

PATCH /api/admin/email-notifications/settings

Description: Update email settings

Body:

```json
{
  "emailNotificationsEnabled": true,
  "videoAnalysisEmailsEnabled": true,
  "maxRetryAttempts": 5,
  "fromEmail": "notifications@yourdomain.com",
  "fromName": "Your App Name"
}
```

Response:

```json
{
  "settings": { ... },
  "message": "Settings updated successfully"
}
```

Troubleshooting

Emails Not Sending

1. **Check API Key**
   - Verify `RESEND_API_KEY` is set in environment
   - Confirm key is valid in Resend dashboard

2. **Check Settings**
   - Navigate to admin settings
   - Ensure `emailNotificationsEnabled` is `true`
   - Ensure `videoAnalysisEmailsEnabled` is `true`

3. **Check Logs**
    - Look for email service logs in console
    - Check for error messages
    - Review failed notification records

4. **Verify Domain**
    - Ensure sending domain is verified in Resend
    - Check DNS records are properly configured

Failed Email Delivery

1. **Check Error Message**
    - View error in admin dashboard
    - Common issues: invalid email, bounced, rate limit

2. **Retry Mechanism**
    - System automatically retries up to 3 times
    - Can manually retry from admin dashboard
    - Check retry count and last retry time

3. **Recipient Issues**
    - Verify recipient email is valid
    - Check if user's mailbox is full
    - Confirm email isn't blocked by recipient

Performance Issues

1. **Async Sending**
    - Emails are sent asynchronously
    - Don't block video analysis completion
    - Failed emails don't break analysis

2. **Rate Limiting**
    - Be aware of Resend rate limits
    - System respects retry delays
    - Monitor sending patterns

Best Practices

For Developers

1. **Always Use Email Service**
    ```typescript
    // ✅ Good
    await emailService.sendEmail({ ... })
    ```

// ❌ Bad - don't call Resend directly
await resend.emails.send({ ... })
```

1. **Handle Errors Gracefully**
    ```typescript
    try {
        await emailService.sendEmail({ ... })
    } catch (error) {
    ```

```
        // Log but don't throw - email failure shouldn't break app
        console.error('Email failed:', error)
    }
```

2. **Use Proper Email Types**

```typescript
    type: 'VIDEO_ANALYSIS_COMPLETE' // ✅
    type: 'CUSTOM' // Use sparingly
```

3. **Track Metadata**

```typescript
    metadata: {
      analysisId: '...',
      score: 85,
      // Useful for debugging
    }
```

For Admins

1. **Monitor Regularly**
   - Check dashboard weekly
   - Review failure rates
   - Address issues promptly

2. **Test Email Settings**
   - Use test emails before major changes
   - Verify domain configuration
   - Check spam folder

3. **Maintain Clean Data**
   - Export and archive old emails periodically
   - Remove invalid email addresses
   - Update settings as needed

Future Enhancements

Planned Features

1. **Email Templates Editor**
   - Visual template designer
   - Custom branding options
   - A/B testing capabilities

2. **Advanced Analytics**
   - Engagement metrics over time
   - User cohort analysis
   - Conversion tracking

3. **More Email Types**
   - Weekly digests
   - Achievement celebrations
   - Training reminders
   - Match recaps

4. **Webhook Integration**
   - Real-time delivery tracking
   - Bounce handling
   - Complaint management

5. **User Preferences**
   - Let users control email frequency
   - Choose notification types
   - Set quiet hours

Support

Documentation

- Email service code: `/lib/email/`
- Templates: `/lib/email/templates/`
- Admin UI: `/app/admin/email-notifications/`
- API routes: `/app/api/admin/email-notifications/`

Resources

- Resend Documentation: https://resend.com/docs
- Prisma Email Schema: `prisma/schema.prisma`
- Email Helpers: `/lib/email/email-helpers.ts`

Contact

For issues or questions about the email system:
1. Check this documentation first
2. Review console logs and admin dashboard
3. Verify Resend configuration
4. Contact development team if issues persist

Last Updated: November 8, 2025
Version: 1.0.0
Status: Production Ready ✅