

AI Integration Setup Guide

This guide will help you set up AI integration for your Outlook AI Assistant. The application supports multiple AI providers and is designed to be easy to configure and customize.

Quick Start

1. Choose Your AI Provider

The application supports three AI providers:

- **OpenAI** (Recommended) - GPT-4, GPT-3.5-turbo
- **Azure OpenAI** - Enterprise-grade OpenAI models
- **Anthropic** - Claude models

2. Get Your API Key

For OpenAI:

1. Go to [OpenAI's website](https://platform.openai.com/) (<https://platform.openai.com/>)
2. Create an account or sign in
3. Navigate to API Keys section
4. Create a new API key
5. Copy the key (starts with `sk-`)

For Azure OpenAI:

1. Set up an Azure OpenAI resource in Azure Portal
2. Get your endpoint URL and API key from the resource
3. Deploy a model (like GPT-4) in Azure OpenAI Studio

For Anthropic:

1. Go to [Anthropic's website](https://console.anthropic.com/) (<https://console.anthropic.com/>)
2. Create an account and get API access
3. Generate an API key

3. Configure Environment Variables

Copy the `.env.example` file to `.env.local` :

```
cp .env.example .env.local
```

Edit `.env.local` and add your AI configuration:

```
# Choose your primary AI provider
AI_PROVIDER=openai

# OpenAI Configuration (if using OpenAI)
OPENAI_API_KEY=your-openai-api-key-here
OPENAI_MODEL=gpt-4

# Azure OpenAI Configuration (if using Azure)
AZURE_OPENAI_API_KEY=your-azure-api-key-here
AZURE_OPENAI_ENDPOINT=https://your-resource.openai.azure.com
AZURE_OPENAI_MODEL=gpt-4

# Anthropic Configuration (if using Anthropic)
ANTHROPIC_API_KEY=your-anthropic-api-key-here
ANTHROPIC_MODEL=claude-3-sonnet-20240229
```

4. Start the Application

```
npm run dev
# or
yarn dev
```

The AI chat interface will now be available in the bottom-right corner of the dashboard!

Detailed Configuration

Environment Variables Reference

| Variable | Description | Default | Required |
|----------------------------|--|---------|-----------------|
| AI_PROVIDER | Primary AI provider (openai , azure-openai , anthropic) | openai | Yes |
| OPENAI_API_KEY | OpenAI API key | - | If using OpenAI |
| OPENAI_MODEL | OpenAI model to use | gpt-4 | No |
| OPENAI_MAX_TOKENS | Maximum tokens per response | 2000 | No |
| OPENAI_TEMPERATURE | Response creativity (0-2) | 0.7 | No |
| AI_RATE_LIMITING_ENABLED | Enable rate limiting | true | No |
| AI_MAX_REQUESTS_PER_MINUTE | Rate limit per minute | 20 | No |
| AI_MAX_REQUESTS_PER_HOUR | Rate limit per hour | 100 | No |

Model Recommendations

OpenAI Models:

- **gpt-4** - Best quality, slower, more expensive
- **gpt-4-turbo** - Good balance of speed and quality
- **gpt-3.5-turbo** - Fastest, cheapest, good for simple tasks

Azure OpenAI Models:

- **gpt-4** - Same as OpenAI GPT-4
- **gpt-35-turbo** - Same as OpenAI GPT-3.5-turbo

Anthropic Models:

- **claude-3-opus-20240229** - Highest capability
- **claude-3-sonnet-20240229** - Balanced performance (recommended)
- **claude-3-haiku-20240307** - Fastest and most cost-effective

Advanced Configuration

Fallback Providers

You can configure fallback providers in case your primary provider fails:

```
AI_FALLBACK_ENABLED=true
AI_FALLBACK_PROVIDERS=openai, azure-openai
```

Rate Limiting

Protect your API usage with built-in rate limiting:

```
AI_RATE_LIMITING_ENABLED=true
AI_MAX_REQUESTS_PER_MINUTE=20
AI_MAX_REQUESTS_PER_HOUR=100
```

Response Caching

Enable caching to reduce API calls for similar requests:

```
AI_CACHING_ENABLED=true
AI_CACHE_TTL_MINUTES=10
```

Customizing AI Behavior

The Prompt Context File

The AI's behavior is controlled by the file `/context/prompt-context.txt`. This file contains detailed instructions that tell the AI how to behave, what it can help with, and how to respond.

Key Sections in the Context File:

1. **Core Identity & Behavior** - Defines the AI's role and personality
2. **Email Management Capabilities** - Instructions for email-related tasks
3. **Calendar & Meeting Management** - Scheduling and calendar guidance
4. **Task & Productivity Management** - Task organization and productivity tips
5. **Safety Guidelines & Limitations** - What the AI can and cannot do
6. **Response Format Guidelines** - How to structure responses

Customizing the AI's Behavior:

To modify how the AI responds, edit `/context/prompt-context.txt`:

```
# Edit the context file
nano /context/prompt-context.txt
```

Example Customizations:

1. **Change the AI's personality:**

```
...
# Make it more formal
You maintain a professional and formal tone at all times.
```

```
# Make it more casual
You use a friendly, casual tone with occasional humor.
...
```

1. Add company-specific information:

```
# Add your company's meeting policies
When scheduling meetings, always remind users that our company policy
requires meetings to be no longer than 50 minutes with 10-minute breaks.
```

2. Customize email templates:

```
# Add your company's email signature format
When drafting emails, always suggest including the company signature:
[Name]
[Title]
[Company Name]
[Contact Information]
```

Response Formatting

The AI is configured to provide structured, helpful responses. You can modify the response format in the context file:

```
### Structure Your Responses
1. Acknowledgment: Briefly acknowledge what the user is asking for
2. Main Content: Provide your primary assistance or advice
3. Additional Options: Offer alternative approaches or related suggestions
4. Next Steps: Suggest concrete actions the user can take
5. Follow-up: Invite further questions or clarification
```

Troubleshooting

Common Issues

1. “AI service is temporarily unavailable”

- **Cause**: API key not configured or invalid
- **Solution**: Check your `.env.local` file and ensure the API key is correct

2. “Too many requests” error

- **Cause**: Rate limiting triggered
- **Solution**: Wait a few minutes or adjust rate limits in environment variables

3. Chat button shows warning icon

- **Cause**: AI service health check failed
- **Solution**: Check your internet connection and API key configuration

4. Responses are too generic

- **Cause**: Context file not being read properly
- **Solution**: Ensure `/context/prompt-context.txt` exists and is readable

Debugging Steps

1. **Check the browser console** for error messages
2. **Verify environment variables** are loaded correctly
3. **Test API connectivity** using the health check endpoint: `/api/ai/chat` (GET request)

4. **Check server logs** for detailed error information

Getting Help

If you're still having issues:

1. Check the application logs in the terminal
2. Verify your API key has sufficient credits/quota
3. Test your API key directly with the provider's API
4. Review the context file for any syntax issues

Security Best Practices

API Key Security

1. **Never commit API keys** to version control
2. **Use environment variables** for all sensitive configuration
3. **Rotate API keys regularly** (every 90 days recommended)
4. **Monitor API usage** to detect unusual activity

Rate Limiting

1. **Keep rate limiting enabled** in production
2. **Monitor usage patterns** to adjust limits appropriately
3. **Set up alerts** for unusual usage spikes

Content Filtering

The application includes basic content filtering to prevent:

- Attempts to bypass AI instructions
- Malicious prompt injection
- Off-topic requests

You can customize these filters in `/lib/ai-service.ts`.

Cost Management

Estimating Costs

- **OpenAI GPT-4**: ~\$0.03 per 1K tokens (input) + \$0.06 per 1K tokens (output)
- **OpenAI GPT-3.5-turbo**: ~\$0.001 per 1K tokens (input) + \$0.002 per 1K tokens (output)
- **Anthropic Claude**: Varies by model, check current pricing

Cost Optimization Tips

1. **Use GPT-3.5-turbo** for simple tasks
2. **Enable caching** to reduce duplicate requests
3. **Set appropriate rate limits** to control usage
4. **Monitor token usage** in the API responses
5. **Optimize the context file** to be concise but comprehensive

Advanced Features

Multiple Provider Setup

You can configure multiple providers and use fallbacks:

```
# Primary provider
AI_PROVIDER=openai
OPENAI_API_KEY=your-openai-key

# Fallback providers
AI_FALLBACK_ENABLED=true
AI_FALLBACK_PROVIDERS=azure-openai,anthropic
AZURE_OPENAI_API_KEY=your-azure-key
AZURE_OPENAI_ENDPOINT=your-azure-endpoint
ANTHROPIC_API_KEY=your-anthropic-key
```

Custom Model Parameters

Fine-tune the AI responses:

```
# More creative responses
OPENAI_TEMPERATURE=1.0

# More focused responses
OPENAI_TEMPERATURE=0.3

# Longer responses
OPENAI_MAX_TOKENS=4000

# Shorter responses
OPENAI_MAX_TOKENS=1000
```

Health Monitoring

The application includes health check endpoints:

- `GET /api/ai/chat` - Check AI service status
- Monitor response times and error rates
- Set up alerts for service degradation

This completes the AI integration setup! Your Outlook AI Assistant now has powerful AI capabilities to help users with email management, calendar scheduling, and productivity optimization.