


















Quicky - AI Summarization Tool

 An intelligent AI-powered summarization tool that transforms various types of content into digestible insights.

Show Image

Features

- **Multi-Format Input Support**
 -  YouTube videos (auto-transcript extraction)
 -  Blog articles and web content
 -  PDF documents and eBooks
 -  Raw text and paragraphs
- **Smart Output Formats**
 -  Bullet points for quick scanning
 -  Structured paragraphs for detailed understanding
 -  Concise notes for reference
 -  Visual mind maps
 -  SEO keywords extraction
 -  Presentation slides format
- **Advanced Features**
 -  Real-time AI processing
 -  Smart caching system
 -  Responsive mobile design
 -  Like/save functionality
 -  One-click copy to clipboard
 -  Regenerate summaries

Technology Stack

- **Backend:** Python Flask
- **Frontend:** HTML5, CSS3, JavaScript
- **Database:** SQLite (dev) / PostgreSQL (prod)
- **AI Integration:** OpenAI GPT, Anthropic Claude, Google Gemini
- **Deployment:** Render.com

Quick Start

Prerequisites

- Python 3.8+
- pip package manager
- Git

Local Development

1. Clone the repository

```
bash
git clone https://github.com/yourusername/quicky-ai-summarizer.git
cd quicky-ai-summarizer
```

2. Set up virtual environment

```
bash
python -m venv venv
source venv/bin/activate # On Windows: venv\Scripts\activate
```

3. Install dependencies

```
bash
pip install -r requirements.txt
```

4. Set up environment variables

```
bash
cp .env.example .env
# Edit .env with your API keys and configuration
```

5. Initialize the database

```
bash
```

```
python init_db.py
```

6. Run the application

```
bash  
  
python quicky_backend.py
```

7. Open your browser

 Navigate to `http://localhost:5000`

Deployment on Render

Automatic Deployment

1. Fork this repository to your GitHub account

2. Connect to Render

- Go to [Render.com](https://render.com)
- Connect your GitHub account
- Select your forked repository

3. Configure Environment Variables

```
FLASK_ENV=production  
SECRET_KEY=your-secret-key  
OPENAI_API_KEY=your-openai-key  
DATABASE_URL=postgresql://... (auto-generated)
```

4. Deploy

- Render will automatically build and deploy your app
- Your app will be available at `https://your-app-name.onrender.com`

Manual Deployment

```
bash  
  
# Build command  
pip install -r requirements.txt  
  
# Start command  
gunicorn --bind 0.0.0.0:$PORT wsgi:app
```

Project Structure

quicky-ai-summarizer/

```
|— quicky_backend.py    # Main Flask application
|— config.py           # Configuration settings
|— wsgi.py             # WSGI entry point
|— init_db.py          # Database initialization
|— content_extractor.py # Content extraction utilities
|— index.html          # Frontend interface
|— frontend_js.js      # Frontend JavaScript
|— requirements.txt     # Python dependencies
|— Procfile            # Process definition for deployment
|— render.yaml         # Render configuration
|— .env.example        # Environment variables template
|— .gitignore          # Git ignore rules
|— README.md           # This file
```

Configuration

Environment Variables

Variable	Description	Required
FLASK_ENV	Environment (development/production)	Yes
SECRET_KEY	Flask secret key for sessions	Yes
DATABASE_URL	Database connection string	Yes
OPENAI_API_KEY	OpenAI API key for GPT models	Optional
ANTHROPIC_API_KEY	Anthropic API key for Claude	Optional
GOOGLE_API_KEY	Google API key for Gemini	Optional
CORS_ORIGINS	Allowed CORS origins	No
UPLOAD_FOLDER	File upload directory	No
MAX_CONTENT_LENGTH	Max file upload size	No

AI API Setup

To use real AI summarization, you need at least one API key:

OpenAI Setup

1. Go to [OpenAI API](#)
2. Create an API key
3. Add to your `.env` file: `OPENAI_API_KEY=sk-...`

Anthropic Claude Setup

- 1. Go to [Anthropic Console](#)
- 2. Create an API key
- 3. Add to your `.env` file: `ANTHROPIC_API_KEY=sk-ant-...`

Google Gemini Setup

- 1. Go to [Google AI Studio](#)
- 2. Create an API key
- 3. Add to your `.env` file: `GOOGLE_API_KEY=...`

Testing

```
bash

# Run basic functionality test
python -c "from quicky_backend import app; print('App imports successfully')"
```

```
# Test database initialization
python init_db.py

# Test with development server
python quicky_backend.py
```

API Endpoints

Endpoint	Method	Description
<code>/</code>	GET	Serve main application
<code>/api/summarize</code>	POST	Generate content summary
<code>/api/upload</code>	POST	Upload file for processing
<code>/api/summary/<id>/like</code>	POST	Like/unlike a summary
<code>/api/summaries/<session></code>	GET	Get session summaries
<code>/api/health</code>	GET	Health check

Usage Examples

Summarize YouTube Video

```
javascript
```

```
POST /api/summarize
{
  "content_type": "video",
  "content_source": "https://youtube.com/watch?v=...",
  "format": "bullets",
  "session_id": "session_123"
}
```

Summarize Blog Article

```
javascript

POST /api/summarize
{
  "content_type": "blog",
  "content_source": "https://example.com/article",
  "format": "paragraphs",
  "session_id": "session_123"
}
```

Upload and Summarize PDF

```
javascript

POST /api/upload
Content-Type: multipart/form-data
file: document.pdf

POST /api/summarize
{
  "content_type": "ebook",
  "content_source": "[extracted_content]",
  "format": "mindmap",
  "session_id": "session_123"
}
```

Contributing

1. Fork the repository
2. Create a feature branch (`git checkout -b feature/amazing-feature`)
3. Commit your changes (`git commit -m 'Add amazing feature'`)

4. Push to the branch (`git push origin feature/amazing-feature`)

5. Open a Pull Request

License

This project is licensed under the MIT License - see the [LICENSE](#) file for details.

Acknowledgments

- [Flask](#) - Web framework
- [OpenAI](#) - AI API provider
- [Anthropic](#) - Claude AI API
- [YouTube Transcript API](#) - Video transcript extraction
- [Beautiful Soup](#) - HTML parsing
- [Render](#) - Deployment platform

Troubleshooting

Common Issues

Database Connection Error

```
bash

# Reset database
python init_db.py --reset
```

Missing Dependencies

```
bash

# Reinstall dependencies
pip install -r requirements.txt --force-reinstall
```




API Key Issues

- Verify your API keys are correctly set in `.env`
- Check API key permissions and billing status
- Ensure the API key format is correct

File Upload Issues

- Check file size limits (default 16MB)
- Verify file format is supported (.pdf, .docx)
- Ensure uploads directory exists and is writable

Support

-  Email: support@quicky-ai.com
-  Issues: [GitHub Issues](#)
-  Discussions: [GitHub Discussions](#)

Made with  for productivity