

Deploying AI Document Assistant to Vercel

This guide provides step-by-step instructions for deploying the AI Document Assistant to Vercel.

Prerequisites

1. A [Vercel](#) account
2. [Git](#) installed on your computer
3. [Node.js](#) installed on your computer
4. An OpenAI API key
5. An ElevenLabs API key (optional, for voice output)

Deployment Steps

1. Prepare the Project

1. Extract the `ai-document-assistant.zip` file to a local directory
2. Navigate to the extracted directory in your terminal

2. Set Up Git Repository

1. Initialize a new Git repository: `bash git init`
2. Add all files to the repository: `bash git add .`
3. Commit the files: `bash git commit -m "Initial commit"`

3. Deploy to Vercel

Option 1: Using Vercel CLI

1. Install the Vercel CLI: `bash npm install -g vercel`
2. Navigate to the `backend` directory: `bash cd backend`
3. Deploy to Vercel: `bash vercel`
4. Follow the prompts to log in and configure your project
5. When asked about the framework, select "Other"
6. Set the build command to: `pip install -r requirements.txt`
7. Set the output directory to: `static`
8. Set the development command to: `python app.py`
9. Set environment variables: `bash vercel env add OPENAI_API_KEY vercel env add ELEVENLABS_API_KEY vercel env add SECRET_KEY`
10. Redeploy with the environment variables: `bash vercel --prod`

Option 2: Using Vercel Web Interface

1. Push your repository to GitHub: `bash # Create a new repository on GitHub first, then: git remote add origin https://github.com/yourusername/ai-document-assistant.git git push -u origin main`
2. Go to [Vercel Dashboard](#)
3. Click "Add New" > "Project"
4. Import your GitHub repository
5. Configure the project:
6. Framework Preset: Other
7. Root Directory: `backend`
8. Build Command: `pip install -r requirements.txt`

9. Output Directory: static
10. Install Command: `pip install -r requirements.txt`
11. Add environment variables:
12. `OPENAI_API_KEY` : Your OpenAI API key
13. `ELEVENLABS_API_KEY` : Your ElevenLabs API key
14. `SECRET_KEY` : A secure random string
15. Click "Deploy"

4. Verify Deployment

1. Once deployment is complete, Vercel will provide a URL for your application
2. Open the URL in your browser
3. Test the application by asking questions about the POH document
4. Verify that voice input and output are working correctly

5. Custom Domain (Optional)

1. In the Vercel dashboard, go to your project settings
2. Click on "Domains"
3. Add your custom domain and follow the instructions to configure DNS

Troubleshooting

Common Issues

1. **Application Error on Deployment**
2. Check the Vercel logs for specific error messages
3. Verify that all environment variables are set correctly
4. Ensure the `requirements.txt` file includes all necessary dependencies
5. **Voice Features Not Working**

6. Check browser permissions for microphone access
7. Verify that the ElevenLabs API key is set correctly
8. Test in a different browser

9. **Document Not Loading**

10. Check that the document is correctly placed in the `documents` directory
11. Verify that the document processing was successful
12. Check the application logs for any errors

Getting Help

If you encounter issues not covered in this guide, please contact support at: - Email: support@redravenllc.ai - Help Desk: help.redravenllc.ai

Next Steps

After successful deployment, you can:

1. Replace the default POH document with your own
2. Customize the application branding
3. Add additional features or integrations

Refer to the `USER_MANUAL.md` and `README.md` files for more information on using and customizing the application.
