# Deployment Documentation for Education App on Atlantic

## Overview

app uses:  
- Frontend: HTML, CSS, JavaScript  
- Backend: Flask (Python)  
- Database: SQLite  
- API: Google Gemini

## Prerequisites

1. Server Configuration:  
 - Atlantic hosting account.  
 - Ensure Atlantic server supports Python 3.7+.  
 - Install basic server dependencies (e.g., SSH access, Python environment setup).  
  
2. Local Files:  
 - Complete app files, including:  
 - frontend folder with HTML, CSS, and JS files.  
 - Flask application files (e.g., app.py, requirements.txt).  
 - SQLite database file.  
  
3. Access Details:  
 - Credentials for Atlantic server.  
 - API Key for Google Gemini.

## Step-by-Step Deployment

### 1. Set Up the Server

1. Access the Atlantic Server:  
 ssh user@your-atlantic-server-ip  
  
2. Update System Packages:  
 sudo apt update && sudo apt upgrade -y  
  
3. Install Python:  
 - Verify Python version:  
 python3 --version  
 - Install pip and virtual environment tools if not present:  
 sudo apt install python3-pip python3-venv -y

### 2. Deploy the Application

1. Create a Project Directory:  
 mkdir ~/education\_app  
 cd ~/education\_app  
  
2. Clone or Transfer Project Files:  
 - Transfer files via SCP or FTP:  
 scp -r /local/path/to/app user@your-atlantic-server-ip:/home/user/education\_app  
  
3. Set Up Virtual Environment:  
 python3 -m venv venv  
 source venv/bin/activate  
  
4. Install Dependencies:  
 pip install -r requirements.txt  
  
5. Set Up Database:  
 - Ensure the SQLite database file is in the correct location.  
 - Verify connectivity:  
 python  
 >>> import sqlite3  
 >>> conn = sqlite3.connect('database\_name.db')  
 >>> print("Database Connected!")  
  
6. Configure API Key:  
 - Add the Google Gemini API key to environment variables or a config file.  
 Example (in app.py):  
 import os  
 os.environ['GEMINI\_API\_KEY'] = 'your\_api\_key\_here'

### 3. Configure Web Server

1. Install Gunicorn:  
 pip install gunicorn  
  
2. Set Up Gunicorn:  
 gunicorn -w 4 -b 0.0.0.0:8000 app:app  
  
3. Install and Configure Nginx:  
 sudo apt install nginx -y  
 - Add the following to the Nginx configuration:  
 server {  
 listen 80;  
 server\_name your-domain.com;  
  
 location / {  
 proxy\_pass http://127.0.0.1:8000;  
 proxy\_set\_header Host $host;  
 proxy\_set\_header X-Real-IP $remote\_addr;  
 proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;  
 }  
  
 location /static/ {  
 root /home/user/education\_app;  
 }  
 }  
 - Restart Nginx:  
 sudo systemctl restart nginx

### 4. Test the Deployment

1. Run the Application:  
 - Test locally:  
 flask run  
 - Access the app in the browser using the server IP or domain.  
  
2. Debugging:  
 - Check logs for errors:  
 tail -f /var/log/nginx/error.log  
 tail -f /var/log/nginx/access.log  
  
 - Test API integration with Google Gemini.