# Sections to include in the document

sections = [

("✨ Project Overview",

"This project demonstrates a full-stack application where:\n"

"- Frontend is built with Node.js + Express\n"

"- Backend is built with Flask\n"

"- Both services are containerized using Docker and connected via Docker Compose"),

("📂 Folder Structure",

"project-root/\n"

"├── frontend/\n"

"│ ├── Dockerfile\n"

"│ ├── package.json\n"

"│ ├── server.js\n"

"│ └── views/\n"

"│ └── form.html\n"

"├── backend/\n"

"│ ├── Dockerfile\n"

"│ ├── app.py\n"

"│ └── requirements.txt\n"

"├── docker-compose.yaml\n"

"└── .gitignore"),

("🛠️ Setup Instructions",

"1. Clone the Repository\n"

"```bash\n"

"git clone <your-repo-url>\n"

"cd <project-folder>\n"

"```\n\n"

"2. Frontend Setup (Node.js + Express)\n"

"package.json:\n"

"{...}\n\n"

"server.js:\n"

"{...}\n\n"

"views/form.html:\n"

"{...}\n\n"

"Dockerfile:\n"

"{...}\n\n"

"3. Backend Setup (Flask)\n"

"app.py:\n"

"{...}\n\n"

"requirements.txt:\n"

"Flask==2.3.2\n\n"

"Dockerfile:\n"

"{...}\n\n"

"4. Docker Compose:\n"

"docker-compose.yaml:\n"

"{...}\n\n"

"5. .gitignore:\n"

"{...}"),

("🚀 Run the Project",

"```bash\n"

"docker-compose up --build\n"

"```\n"

"- Frontend: http://localhost:3000\n"

"- Backend: Accessible via internal network only"),

("🛳 Push Docker Images to Docker Hub",

"```bash\n"

"docker build -t yourdockerhubusername/frontend ./frontend\n"

"docker build -t yourdockerhubusername/backend ./backend\n"

"docker login\n"

"docker push yourdockerhubusername/frontend\n"

"docker push yourdockerhubusername/backend\n"

"```"),

("💾 GitHub Push",

"```bash\n"

"git init\n"

"git remote add origin https://github.com/<yourusername>/<repo-name>.git\n"

"git add .\n"

"git commit -m \"Initial commit\"\n"

"git push -u origin main\n"

"```"),

("📄 Submission Checklist",

"- [x] Frontend and backend structured\n"

"- [x] Dockerfiles written\n"

"- [x] Docker Compose file created\n"

"- [x] Pushed images to Docker Hub\n"

"- [x] Uploaded project to GitHub\n"

"- [x] .gitignore added\n\n"

"Done! 🚀👍")

]

# Add sections to the document

for heading, content in sections:

doc.add\_heading(heading, level=2)

for paragraph in content.split('\n'):

doc.add\_paragraph(paragraph)

# Save the Word document

doc\_path = "/mnt/data/Node-Flask-Docker-Project.docx"

doc.save(doc\_path)

doc\_path

import zipfile

import os

# Define the folder structure and files

project\_structure = {

"frontend/": {

"Dockerfile": """FROM node:18

WORKDIR /app

COPY . .

RUN npm install

EXPOSE 3000

CMD ["node", "server.js"]""",

"package.json": """{

"name": "frontend",

"version": "1.0.0",

"main": "server.js",

"dependencies": {

"express": "^4.18.2",

"body-parser": "^1.20.2",

"axios": "^1.6.7"

}

}""",

"server.js": """const express = require('express');

const bodyParser = require('body-parser');

const path = require('path');

const axios = require('axios');

const app = express();

app.use(bodyParser.urlencoded({ extended: true }));

app.use(express.static('views'));

app.get('/', (req, res) => {

res.sendFile(path.join(\_\_dirname, 'views', 'form.html'));

});

app.post('/submit', (req, res) => {

const data = req.body;

axios.post('http://backend:5000/submit', data)

.then(response => res.send(response.data))

.catch(error => res.send('Error sending data to backend: ' + error));

});

app.listen(3000, () => {

console.log('Frontend running on http://localhost:3000');

});""",

"views/form.html": """<!DOCTYPE html>

<html>

<body>

<h2>Assignment Form</h2>

<form action="/submit" method="post">

<label>Name:</label>

<input type="text" name="name" required><br><br>

<label>Email:</label>

<input type="email" name="email" required><br><br>

<input type="submit" value="Submit">

</form>

</body>

</html>"""

},

"backend/": {

"Dockerfile": """FROM python:3.10

WORKDIR /app

COPY . .

RUN pip install -r requirements.txt

EXPOSE 5000

CMD ["python", "app.py"]""",

"app.py": """from flask import Flask, request

app = Flask(\_\_name\_\_)

@app.route('/submit', methods=['POST'])

def submit():

name = request.form.get('name')

email = request.form.get('email')

return f"Received from Frontend: Name={name}, Email={email}"

if \_\_name\_\_ == '\_\_main\_\_':

app.run(host='0.0.0.0', port=5000)""",

"requirements.txt": "Flask==2.3.2"

},

"": {

"docker-compose.yaml": """version: "3.8"

services:

frontend:

build: ./frontend

ports:

- "3000:3000"

depends\_on:

- backend

networks:

- app-network

backend:

build: ./backend

ports:

- "5000:5000"

networks:

- app-network

networks:

app-network:

driver: bridge""",

".gitignore": """# Node.js

dist/

node\_modules/

npm-debug.log

# Python

\_\_pycache\_\_/

\*.pyc

# IDE/Editor

.vscode/

.env

.DS\_Store"""

}

}