Documentation Tool: Swagger (OpenAPI)

Introduction:

Swagger (based on the OpenAPI Specification) is a set of tools that helps developers design, build, document, and consume REST APIs. It provides an interactive documentation interface, allowing both developers and non-developers to explore and test API endpoints in real-time.

How Swagger Works:

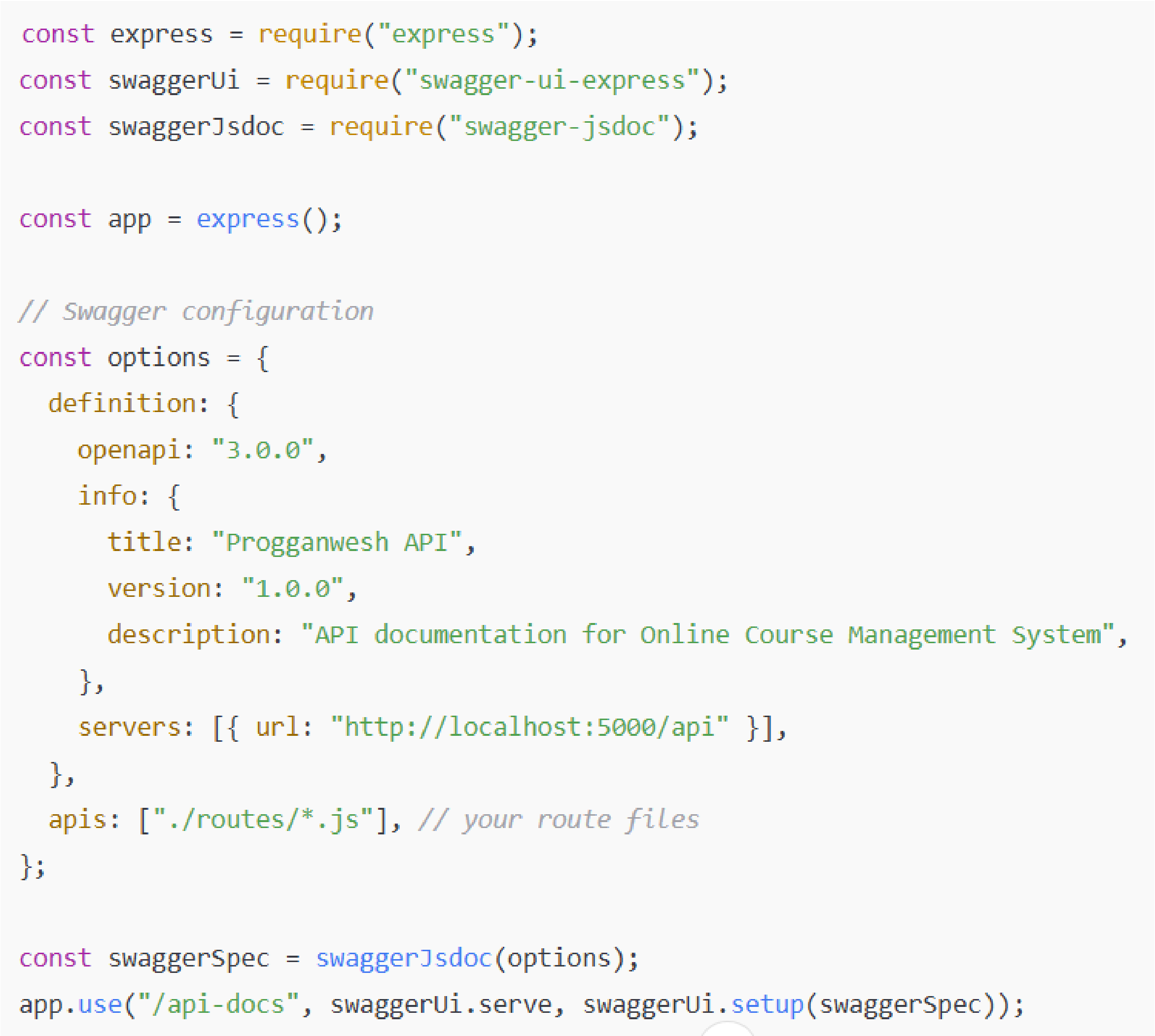
1. Write an API specification (in JSON or YAML) that defines endpoints, methods, parameters, and responses.
2. Use Swagger UI to render the spec as a dynamic documentation site.
3. Test API endpoints directly from the documentation.
4. Keep frontend and backend aligned with a single source of truth.

Basic Syntax Example (YAML):

```
openapi: 3.0.0
info:
  title: Progganwesh API
  version: 1.0.0
  description: API for Online Course Management System
servers:
  - url: http://localhost:5000/api

paths:
  /login:
    post:
      summary: User login
      requestBody:
        required: true
        content:
          application/json:
            schema:
              type: object
              properties:
                email:
                  type: string
                  example: student@example.com
                password:
                  type: string
```

```yaml
                example: pass123
    responses:
      '200':
        description: Login successful
      '401':
        description: Invalid credentials

/courses:
  get:
    summary: Get all courses
    responses:
      '200':
        description: List of courses

/courses/{id}:
  get:
    summary: Get course by ID
    parameters:
      - name: id
        in: path
        required: true
        schema:
          type: integer
    responses:
      '200':
        description: Course details
      '404':
        description: Course not found
```

Swagger Setup in Your Server File



```
const express = require("express");
const swaggerUi = require("swagger-ui-express");
const swaggerJsdoc = require("swagger-jsdoc");

const app = express();

// Swagger configuration
const options = {
  definition: {
    openapi: "3.0.0",
    info: {
      title: "Progganwesh API",
      version: "1.0.0",
      description: "API documentation for Online Course Management System",
    },
    servers: [{ url: "http://localhost:5000/api" }],
  },
  apis: ["./routes/*.js"], // your route files
};

const swaggerSpec = swaggerJsdoc(options);
app.use("/api-docs", swaggerUi.serve, swaggerUi.setup(swaggerSpec));
```

Installation:

Swagger can be used through Swagger Editor (web-based), SwaggerHub (hosted solution), or integrated into a Node.js project with libraries such as swagger-ui-express and swagger-jsdoc.

To install in a Node.js project: npm install

swagger-ui-express swagger-jsdoc

Usage:

1. Define your API specification using comments or YAML/JSON files. 2. Integrate Swagger UI in your Node.js server to serve interactive documentation. 3. Access your documentation via http://localhost:5000/api-docs (or your configured path).

Advantages of Swagger:

- Interactive documentation with live testing.
- Widely adopted industry standard.
- Improves collaboration between frontend and backend teams.
- Supports versioning, authentication, and advanced use cases.

Disadvantages of Swagger:

- Requires extra setup effort compared to inline documentation tools.
- The specification must be updated when APIs change, otherwise it becomes outdated.
- Learning curve for those new to OpenAPI specification.

Summary:

Swagger (OpenAPI) is a powerful and widely used tool for API documentation. It provides interactive, real-time testing of endpoints and acts as a single source of truth for backend and frontend teams. While it requires setup and maintenance, its benefits for clarity, professionalism, and collaboration make it an excellent choice for projects like Progganwesh.