MERN Stack Assignment 3

1. **What features does Swagger offer?**

   Swagger is a set of open-source tools built around the OpenAPI Specification that can help you design, build, document and consume REST APIs. The major Swagger tools include:

   Swagger Editor – browser-based editor where you can write OpenAPI specs.
   Swagger UI – renders OpenAPI specs as interactive API documentation.
   Swagger Codegen – generates server stubs and client libraries from an OpenAPI spec.

2. **How does API testing benefit you?**

   API testing is a type of software testing that verifies Application Programming Interfaces—often referred to as APIs. API testing confirms that an application's performance, functionality, security and reliability are performing as expected. QA teams enjoy the benefits of API automation when executing test cases with the help of API testing tools.

   If you are thinking about the need for API testing, the following section will help you. We are living in a time where users expect everything ready. This is the reason; IT companies are inclining towards the concept of microservices.

   Moreover, microservices facilitate different datastores corresponding to each section of the application that requires different commands for operations.

   Companies prefer to use the concept of microservices because it allows quick deployment which further makes the development process smoother.

   However, we have shared the concept of microservices because APIs play an important role here. Each section of the applications gets commands through the API only. Now you must be able to connect microservices and APIs.

   Hence, API testing is a must to do because it helps to identify the errors or bugs at the very early stage of development.

3. **How do we include an API in Swagger?**

   To include an API in Swagger using YAML:
   First, we have to create a YAML file in which we will write the code like:
   paths:
   /:
   get:
   summary:
   description:
   responses:
   here '/' is the endpoint and 'get' is the method.
   We can change these things as per the requirement.

4. **In Swagger UI, how can we submit custom headers with requests?**

In Swagger, API operation parameters are defined under the parameters section in the operation definition. Each parameter has name, value type or schema (for request body), and optional description. Here is an example:

```
paths:
  /users/{userId}:
    get:
      summary: Gets a user by ID.
      parameters:
        - in: path
          name: userId
          type: integer
          required: true
          description: Numeric ID of the user to get.
```

Note that parameters is an array, so, in YAML, each parameter definition must be listed with a dash (-) in front of it.

5. **How should we use Swagger to produce static docs?**

To get an interactive static page like the Swagger UI, follow these steps:

Install
Go to https://github.com/swagger-api/swagger-ui/releases and download the latest release as a .zip file.
Unzip the file and copy everything in the ./dist folder over to the directory that you want the webserver to serve up. For example, Gitlab Pages needs it needs to be in the ./public folder in your repository.
Copy over your swagger.yml file to the ./public folder.
Open up the ./public/index.html file and update the URL to where the swagger file will be on the webserver. For a local server, it might be this: url: "http://localhost:8000/swagger.yml
Test
To test this out, you can run a simple HTTP server using python3.

```
python3 -m http.server 8000 --directory public
```

Open up http://localhost:8000/ and check it out!