Static Design Documentation for Test Cases:

GetClients API

1. Test to get all clients successfully:
   - Input: GET request to the API endpoint to get all clients
   - Expected output: HTTP status code 200 and a JSON array of all clients
2. Test to handle errors in getting clients:
   - Input: GET request to the API endpoint to get all clients when there is an error in getting the clients (e.g. the database is down)
   - Expected output: HTTP status code 500 and a JSON error message indicating that there was an error in getting the clients
3. Test to verify the Content-Type header:
   - Input: GET request to the API endpoint to get all clients
   - Expected output: HTTP status code 200 and a JSON array of all clients with the Content-Type header set to "application/json"

GetClient API

1. Test to retrieve an existing client:
   - Input: id parameter of an existing client
   - Expected output: HTTP status code 200 and the JSON representation of the client
2. Test to retrieve a non-existing client:
   - Input: id parameter of a non-existing client
   - Expected output: HTTP status code 404 with an error message indicating that the client was not found
3. Test to retrieve a client with an invalid ID parameter:
   - Input: id parameter that is not in the correct format (e.g. a string instead of an integer)
   - Expected output: HTTP status code 400 with an error message indicating that the ID parameter is invalid
4. Test to retrieve a client with a missing ID parameter:
   - Input: Request URL without the id parameter
   - Expected output: HTTP status code 400 with an error message indicating that the ID parameter is missing

5. Test to verify the Content-Type header:
    - Input: Request to retrieve an existing client
    - Expected output: HTTP status code 200 and the JSON representation of the client with the Content-Type header set to "application/json"

CreateClient API

1. Test to create a new client successfully:
    - Input: POST request to the API endpoint with a valid JSON object representing a new client
    - Expected output: HTTP status code 200 and a JSON object representing the newly created client
2. Test to handle errors in decoding request body:
    - Input: POST request to the API endpoint with an invalid JSON object representing a new client (e.g. missing required fields)
    - Expected output: HTTP status code 400 and a JSON error message indicating that the request body is invalid
3. Test to handle errors in adding a new client:
    - Input: POST request to the API endpoint with a valid JSON object representing a new client, but adding the client fails due to some error (e.g. database is down)
    - Expected output: HTTP status code 500 and a JSON error message indicating that the client could not be created

UpdateClient API

1. Test to update an existing client with valid data:
    - Input: id parameter of an existing client and a valid JSON representation of the updated client data in the request body
    - Expected output: HTTP status code 200 and the updated JSON representation of the client
2. Test to update a non-existing client:
    - Input: id parameter of a non-existing client and a valid JSON representation of the updated client data in the request body
    - Expected output: HTTP status code 404 with an error message indicating that the client was not found
3. Test to update a client with an invalid ID parameter:

    - Input: id parameter that is not in the correct format (e.g. a string instead of an integer) and a valid JSON representation of the updated client data in the request body
    - Expected output: HTTP status code 400 with an error message indicating that the ID parameter is invalid

4. Test to update a client with a missing ID parameter:
    - Input: Request URL without the id parameter and a valid JSON representation of the updated client data in the request body
    - Expected output: HTTP status code 400 with an error message indicating that the ID parameter is missing
5. Test to update a client with invalid JSON data in the request body:
    - Input: id parameter of an existing client and invalid JSON data in the request body
    - Expected output: HTTP status code 400 with an error message indicating that the request body is invalid
6. Test to verify the Content-Type header:
    - Input: Request to update an existing client with valid data
    - Expected output: HTTP status code 200 and the updated JSON representation of the client with the Content-Type header set to "application/json"

DeleteClient API

1. Test to delete an existing client successfully:
    - Input: DELETE request to the API endpoint with the ID of an existing client
    - Expected output: HTTP status code 204 and a JSON message indicating that the client was deleted successfully
2. Test to handle errors when attempting to delete a non-existent client:
    - Input: DELETE request to the API endpoint with the ID of a client that does not exist
    - Expected output: HTTP status code 404 and a JSON error message indicating that the client was not found
3. Test to handle errors in decoding request parameters:
    - Input: DELETE request to the API endpoint with an invalid parameter value (e.g. a non-numeric value)
    - Expected output: HTTP status code 400 and a JSON error message indicating that the request parameter is invalid.