Trello Practice Peer-graded Assignment: REST Services

The following details draw attention to Trello's affirmation of the restrictions and principles of REST Design:

Client-Server Architecture: When a user interacts with the user interface, the web application sends requests to the server. The client can ingest and handle the server's relevant response, which includes HTTP Status codes and response messages, as needed.

Layered: Trello's architecture is layered, with the Presentation Layer, Application Layer, and Data Layer all separated from one another. Trello was able to accomplish separation of concerns and modularity because of this.

Stateless: Each request made through the Trello UI is distinct and self-contained. Any previous queries made by the browser are not required to be stored by the server. Trello Client demonstrates that the implementation is stateless by sending cookies and requests as authentication data.

Cacheable: The server's HTTP response includes Cache-Control headers with various max-age values. This demonstrates that the resources the server responded to are cacheable.

Uniform Interface: The request and response bodies are written in JSON format, URIs identify the resources, and requests are sent using conventional HTTP methods, including GET, POST, PUT, and DELETE. This structure is followed throughout the entire application.

By Rasool Hashemi on 9-Jan-2023