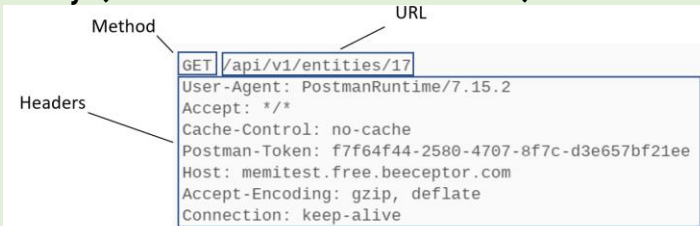
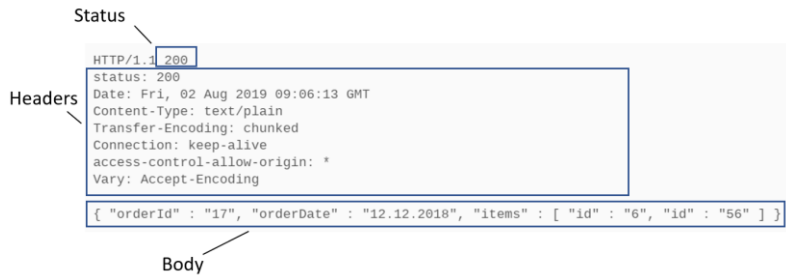


Software Architecture Series: REST API Design

Check List

When designing your REST API, make sure to follow all the following items. Doing that will ensure your REST API is professional, easy to use, and attractive.

Item	Description	Comments																				
Request	<div><div>The request should include:</div><ul style="list-style-type: none">- Method- URL- Headers- Body (if Method is POST or PUT)</div>																					
Response	<div><div>The response should include:</div><ul style="list-style-type: none">- Status Code- Headers- Body (Optional)</div>																					
HTTP Verb	<table><tr><th>Verb</th><th>Role</th><th>Body?</th><th>Params In...</th></tr><tr><td>GET</td><td>Retrieve resource(s)</td><td>No</td><td>URL</td></tr><tr><td>POST</td><td>Add resource(s)</td><td>Yes</td><td>Body</td></tr><tr><td>PUT</td><td>Modify resource(s)</td><td>Yes</td><td>Body</td></tr><tr><td>DELETE</td><td>Delete resource(s)</td><td>No</td><td>URL</td></tr></table>	Verb	Role	Body?	Params In...	GET	Retrieve resource(s)	No	URL	POST	Add resource(s)	Yes	Body	PUT	Modify resource(s)	Yes	Body	DELETE	Delete resource(s)	No	URL	<div>Other, rarely used verbs: PATCH – Partial updates HEAD – GET without the body</div>
Verb	Role	Body?	Params In...																			
GET	Retrieve resource(s)	No	URL																			
POST	Add resource(s)	Yes	Body																			
PUT	Modify resource(s)	Yes	Body																			
DELETE	Delete resource(s)	No	URL																			

URL Structure	<div><div>API Word</div><div>Entity</div><div>Sub Entity</div><div>Version</div><div>ID Param</div><div>Query Parameters</div></div> <div>/api/v1/order/17/items?user=john&date=12/12/2018</div>	<ul style="list-style-type: none">- NEVER include verbs in URL- Some parts are optional (ie. Sub-entity)- Use singular/plural wisely										
Response Codes	<table><thead><tr><th>Verb</th><th>Allowed Codes</th></tr></thead><tbody><tr><td>GET</td><td>200 OK 400 Bad Request 401 / 403 Authorization Related 404 Not Found</td></tr><tr><td>POST</td><td>200 OK 201 Created 202 Accepted 204 No Content 400 Bad Request 401 / 403 Authorization Related 404 Not Found</td></tr><tr><td>PUT</td><td>200 OK 202 Accepted 204 No Content 400 Bad Request 401 / 403 Authorization Related 404 Not Found</td></tr><tr><td>DELETE</td><td>200 OK 400 Bad Request 401 / 403 Authorization Related 404 Not Found</td></tr></tbody></table>	Verb	Allowed Codes	GET	200 OK 400 Bad Request 401 / 403 Authorization Related 404 Not Found	POST	200 OK 201 Created 202 Accepted 204 No Content 400 Bad Request 401 / 403 Authorization Related 404 Not Found	PUT	200 OK 202 Accepted 204 No Content 400 Bad Request 401 / 403 Authorization Related 404 Not Found	DELETE	200 OK 400 Bad Request 401 / 403 Authorization Related 404 Not Found	<ul style="list-style-type: none">- Avoid non-standard codes- Consider use of 207 Multi Status code when necessary
	Verb	Allowed Codes										
	GET	200 OK 400 Bad Request 401 / 403 Authorization Related 404 Not Found										
	POST	200 OK 201 Created 202 Accepted 204 No Content 400 Bad Request 401 / 403 Authorization Related 404 Not Found										
	PUT	200 OK 202 Accepted 204 No Content 400 Bad Request 401 / 403 Authorization Related 404 Not Found										
DELETE	200 OK 400 Bad Request 401 / 403 Authorization Related 404 Not Found											
Docs	<ul style="list-style-type: none">- ALWAYS use OpenAPI / Swagger for documenting the API- Great for testing and documentation- Doesn't have to be exposed to external clients											
Versioning	<ul style="list-style-type: none">- Include version in the API (URL or Header)- Define version support policy											
Authn & Authz	<ul style="list-style-type: none">- Prefer using widely-used protocols such as OAuth- Use popular authentication servers if possible	If the API is internal and cannot utilize external authn servers, check if local AD can be used.										
Performance	<ul style="list-style-type: none">- If needed, make sure Rate Limit and Quota are in place- Use caching whenever possible											

Monitoring	<ul style="list-style-type: none">- Collect API usage data- Use this data to improve the API	DO NOT develop your own monitoring engine, Use a 3 rd party for that.
------------	---	--

Hope you enjoyed the course, and that it helped you creating great REST API!

For any question or comment contact me at:

memi@memilavi.com

Thanks,

Memi