

Concerter Service  
A RESTful API Documentation

Bogazici University  
Fundamentals of Software Engineering

Group 9

## 1. What is RESTful ?

Representational state transfer or RESTful Web services are one way of providing interoperability between computer systems on the Internet. This architectural style for reliable Web applications known as Representation State Transfer (REST) has been proposed by Roy Fielding and has inspired both the W3C Technical Architecture Group's architecture document Web Arch and many who see it as a model for how to build Web services. REST is a hybrid style derived from several of the network-based architectural styles and combined with additional constraints that define a uniform connector interface.

The HTTP verbs comprise a major portion of our “uniform interface” constraint and provide us the action counterpart to the noun-based resource. The primary or most-commonly-used HTTP methods are POST, GET, PUT, PATCH, and DELETE. These correspond to create, read, update, and delete (or CRUD) operations, respectively. There are several other verbs, too, but are utilized less frequently. Of those less-frequent methods, OPTIONS and HEAD are used more often than others.

HTTP VERB: GET

Crud operation: READ

Returns: 200 (OK), list of requested entities or 404 (Not Found) if request is not found or invalid request.

HTTP VERB: POST

Crud operation: Write

Returns: 201 (Created), 404 (Not Found) or, 409 (Conflict) if requested resource already exists.

## HTTP VERB: PUT

Crud operation: Update

Returns:

- Positive: 200 (OK) or 204 (No content). 204 status code is received if the response contains a header field, then the PUT was successful and the header field value contains the entity-tag for the new representation of that target resource.
- Negative: 404 (Not Found), 405 (Method not allowed). 405 status code is received when using PUT on a read-only resource.

## HTTP VERB: PATCH

Crud operation: Update

Returns:

- Positive: 200 (OK) or 204 (No Content)
- Negative: 404 (Not Found) or, 405 (Method not allowed).

## HTTP VERB: DELETE

Crud operation: Delete

Returns:

- Positive: 200 (OK). When the entity is successfully deleted.
- Negative: 404 (Not Found) or, 405 (Method not allowed).

## 2. Concerter Service Documentation

Function	Write a comment
URL	/comment
Method	POST
Data Parameters	<pre>{   ownerID : [integer],   upVote : [integer],   downVote : [integer]   content: [string] }</pre>
Success Response	201 (Created)
Error Response	Code: 500 (Internal Server Error) Meaning: There might be an error in sent JSON.

Function	Get All Concerts
URL	/concert
Method	GET
Data Parameters	None.
Success Response	<pre>[   {     id : [integer],     name : [string],     ownerId : [integer],     artistName : [string],     locationX = [double],     LocationY = [double],     minPrice = [double],     maxPrice = [double],     rate = [double]   },   {     ...   } ]</pre> <p>HTTP 200 (OK)</p>
Error Response	<p>Code: 404 (Not Found)</p> <p>Meaning: No concert entity is specified on the database.</p>

Function	Get all comments
URL	/comment
Method	GET
Data Parameters	None.
Success Response	<pre>{   ownerID : [integer],   upVote : [integer],   downVote : [integer]   content: [string], }</pre> <p>200 (OK)</p>
Error Response	<p>Code: 404 (Not Found)</p> <p>Meaning: No comment entity is specified on the database.</p>

Function	Login
URL	/user
Method	GET
Data Parameters	email : [string], password: [string]
Success Response	<pre> {     id : [integer],     facebook_id : [integer],     google_id : [integer],     name : [string],     username : [string],     email : [string],     password : [string] } 200 (OK) </pre>
Error Response	Code: 404 (Not Found) Meaning: No user entity is specified on the database.

Function	Sign Up
URL	/user
Method	POST
Data Parameters	<pre>{   name : [string],   username: [string],   email : [string],   password: [string] }</pre>
Success Response	201 (Created)
Error Response	<p>Code: 500 (Internal Server Error)</p> <p>Meaning: There might be an error in sent JSON.</p>



Function	Get a concert
URL	/concert/<id>
Method	GET
Data Parameters	id : int
Success Response	<pre>{   id : [integer],   name : [string],   ownerId : [integer],   artistName : [string],   locationX = [double],   LocationY = [double],   minPrice = [double],   maxPrice = [double],   rate = [double] },</pre> <p>200 (OK)</p>
Error Response	<p>Code: 404 (Not Found)</p> <p>Meaning: No user entity is specified on the database.</p>

Function	Add a concert
URL	/concert
Method	POST
Data Parameters	<pre>{   id : [integer],   name : [string],   ownerId : [integer],   artistName : [string],   locationX = [double],   LocationY = [double],   minPrice = [double],   maxPrice = [double],   rate = [double] },</pre>
Success Response	201 (Created)
Error Response	Code: 500 (Internal Server Error) Meaning: There might be an error in sent JSON.

### 3. References

<https://httpstatusdogs.com/405-method-not-allowed>

<http://www.restapitutorial.com/lessons/httpmethods.html>

<https://www.w3.org/TR/2004/NOTE-ws-arch-20040211/#relwwwrest>

<http://www.checkupdown.com/status/E405.html>

<https://bocoup.com/blog/documenting-your-api>