

# Kwik Koder Rest

URL: <https://kwickkoderrest.azurewebsites.net/>

## Users

Endpoint	Input	Output	Requires Authorization	HTTP
/api/User/username	None	{ "userName": "string", "name": "string", "userId": 0, "revapoints": 0 }	Yes	GET
/api/UserStat/all	None	[ { "userID": "string", "averagewpm": 0, "averageaccuracy": 0, "numberoftests": 0, "totaltesttime": 0, "category": 0 } ]	Yes	GET
/api/UserStat/tests	None	[ { "numberofcharacters": 0, "numberoferrors": 0, "wpm": 0, "timetakenms": 0, "date": "2021-06-18T00:12:57.845Z" } ]	Yes	GET
/api/UserStat	None	{ "userID": "string", "averagewpm": 0, "averageaccuracy": 0, "numberoftests": 0, "totaltesttime": 0, "category": 0 }	Yes	GET

## Details:

These endpoints read the current user's Auth0Id and uses it to find their information in the database.

**GET /api/User/username :**

Used to get user's username and name from Auth0 as well as their Id in the database and revapoints.

Returns: Output or a 404 if user is not found.

**GET /api/UserStat/all :**

Used to get current user's performance in all categories as an Ienumerable with each average performance as a separate item.

Returns: Output or a 404 if statistics are not found.

**GET /api/UserStat/tests:**

Used to get current user's latest 100 tests as a list of tests sorted by date (oldest to newest).

Returns: Output or a 404 if user is not found.

**GET /api/UserStat:**

Used to get user's average statistics, category returned is set to be user's revapoints.

Returns: Output or a 404 if statistic is not found.

[Leaderboard](#)

Endpoint	Input	Output	Requires Authorization	HTTP
/api/LB	None	[ { "userName": "string", "name": "string", "averageWPM": 0, "averageAcc": 0, "ranking": 0 } ]	No	GET
/api/LB/{id}	URL: int id, Refers to category number	[ { "userName": "string", "name": "string", "averageWPM": 0, "averageAcc": 0, "ranking": 0 } ]	No	GET

#### Details:

These endpoints are used to get the users ranked by their performance.

#### GET /api/LB:

Used to get the users sorted by their overall WPM as a list with their ranking and information.

Returns: Output or 404 if leaderboard cannot be found.

#### GET /api/LB/{id}:

Used to get the users sorted by their overall WPM as a list with their ranking and information.

Id is either -1 for quotes or the number of the language in the Octokit.Language Reference, see Appendix for more details.

Returns: Output or 404 if leaderboard cannot be found.

#### Tests

Endpoint	Input	Output	Requires Authorization	HTTP
/api/TypeTest	None	{ "content": "string", "author": "string", "length": 0, "categoryId": 0 }	No	GET
/api/TypeTest/{id}	URL: int id, Refers to category number	{ "content": "string", "author": "string", "length": 0, "categoryId": 0 }	No	GET
/api/TypeTest	{ "categoryId": 0, "numberOfcharacters": 0, "numberOferrors": 0, "wpm": 0, "timetakenms": 0, "date": "2021-06-18T01:38:51.924Z" }	None	Yes	POST

#### Details:

These endpoints are used for getting quotes for test as well as posting results for tests

#### GET /api/TypeTest:

Used to get a random quote, author, and length from <http://api.quotable.io/random>

Returns: Output or 500 on internal server error.

#### **GET /api/TypeTest/{id}:**

Used to get a random quote on -1 or language number from Octokit.Language to search in <https://raw.githubusercontent.com/> for a random file in that language to get for coding test.

Gets FullName from the repository as "author," file content as "content," length as length, and the "categoryId" as the language number from Octokit.Language

Returns: Output or 500 on internal server error.

#### **POST /api/TypeTest:**

Used to post the results of a type test to the database, adding a category and/or user if necessary and updating user stats.

Returns: 200 or 400 if given a bad input.

### Competitions

Endpoint	Input	Output	Requires Authorization	HTTP
/api/Competition	None	[ { "name": "string", "start": "2021-06-18T02:21:56.857Z", "end": "2021-06-18T02:21:56.857Z", "category": 0, "snippet": "string", "author": "string", "compId": 0 } ]	No	GET
/api/Competition/{id}	URL: int id, Refers to id of competition	[ { "wpm": 0, "rank": 0, "userName": "string", "accuracy": 0, "name": "string", "compName": "string", "userId": 0 } ]	No	GET

/api/Competition	{ "name": "string", "start": "2021-06-18T02:39:36.442Z", "end": "2021-06-18T02:39:36.442Z", "category": 0, "snippet": "string", "author": "string", "compld": 0 }	link to new competition and competition Id as int in body	Yes	POST
/api/Competition/bet	{ "userBetOn": 0, "compld": 0, "betAmount": 0 }	None	Yes	PUT
/api/Competition/bet/{id}	URL: int id, Refers to id of user Body: any	None	Yes	PUT

#### Details:

These endpoints, deal with creating/viewing/viewing results of/betting on competitions. The Competition Stats are more to deal with directly accessing and posting the relevant user test.

#### **GET /api/Competition:**

Used to get a list of all competitions in the database with relevant Dates for start and end, as well as information around the competition.

Returns: Output or 404 if the competitions cannot be found.

#### **GET /api/Competition/{id}:**

Used to get the result of a competition as a list of the participants' entries.

Returns: Output or 404 if no such competition with the id can be found.

#### **POST /api/Competition:**

Used to post a new competition to the database.

Returns: 201 with route and then new competition Id in body, otherwise 400 if user tries to enter bad data.

#### **PUT /api/Competition/bet:**

Used to put a bet into the database for the currently logged in user. Places bet based on competition, userId, and amount you give it.

Returns: 200 or 400 if betting on a finished competition, if you do not have enough revapoints, or if you haven't participated in a test yet.

#### **PUT /api/Competition/bet/{id}**

Method that updates user's revapoints given their id and claims their winnings, adding them to their overall points.

Returns: 200 or 404 if no relevant bets can be found to claim.

### Competition Stats

Endpoint	Input	Output	Requires Authorization	HTTP
/api/CompetitonStats/{id}	URL: int id, Refers to id of competition	{ " id": 0, "testString": "string", "author": "string", "categoryId": 0 }	No	GET
/api/CompetitonStats	{ "categoryId": 0, "numberOfcharacters": 0, "numberOferrors": 0, "wpm": 0, "timetakenms": 0, "date": "2021-06- 18T03:36:48.833Z", "compId": 0 }	New ranking as int in body	Yes	POST

#### Details:

These endpoints directly relate to the test that the competition is based around, providing it or posting the new result.

#### **GET /api/CompetitionStats/{id}:**

Used to get the information needed for a user to take the test in the competition and have the results affect their profile statistics.

Returns: Output or 404 if competition cannot be found.

**POST /api/CompetitionStats:**

Used to post the user's competition result to the database and updates their statistics accordingly.

Returns: 200 with rank in the body or 400 if the test input is incorrect or the competition is already done. Returns 404 if competition cannot be found.

## Appendix: Category Naming System

For the list of currently supported languages see the enum below:

{	Quotes = -1,	ActionScript = 1,
Ada = 2,	AppleScript = 4,	Arduino = 6,
Asp = 8,	Awk = 12,	C = 18,
Cobol = 24,	Cmake = 25,	CoffeeScript = 27,
CPlusPlus = 31,	CSharp = 32,	Css = 33,
CppObjDump = 34,	Cython = 36,	EmacsLisp = 51,
FSharp = 53,	Fortran = 59,	Gsl = 65,
Go = 66,	Haskell = 73,	Java = 81,
JavaScript = 83,	Json = 84,	Kotlin = 87,
Less = 89,	Llvm = 95,	Lua = 98,
Makefile = 100,	Markdown = 102,	Matlab = 103,
Moonscript = 109,	NumPY = 117,	ObjectiveC = 119,
ObjectiveJ = 120,	OpenCL = 125,	Pascal = 130,
Perl = 131,	Php = 132,	PowerShell = 135,
Python = 140,	R = 142,	Racket = 143,
Ruby = 151,	Rust = 152,	Scala = 153,
Sass = 156,	Scss = 158,	Shell = 160,
Squirrel = 164,	SuperCollider = 166,	TeX = 170,
TypeScript = 176,	Verilog = 180,	Vhdl = 181,
VimL = 182,	VisualBasic = 183,	Volt = 184,
Wisp = 185,	Xml = 187,	XQuery = 189,
Yaml = 193	}	