



Let's Multiply! ^{1.0.0}

Capstone Week 3 - Design Service Layer Re-Design

[Terms of service](#)

[Contact the developer](#)

[Apache 2.0](#)

Schemes

HTTP

user



GET /login Logs user into the application.

Username and password combination must match and be verified through the database.

Parameters

Try it out

| Name | Description |
|----------------------------|---|
| username * required | The user name for login |
| string | |
| (query) | username - The user name for login |
| password * required | The password for login in clear text |
| string | |
| (query) | password - The password for login in clear text |

Responses

Response content type

application/xml

Code

Description

200

Login successful

Example Value Model

```
<?xml version="1.0" encoding="UTF-8"?>
<UserData>
  <_id>123456789</_id>
  <firstName>Test Student FN</firstName>
  <lastName>Test Student LN</lastName>
  <isStudent>true</isStudent>
  <isParent>false</isParent>
  <isTeacher>false</isTeacher>
</UserData>
```

400

Something went wrong

401

User not authorized

GET

/logout Logs out current logged in user session

Parameters

Try it out

No parameters

Responses

Response content type

application/xml

Code

Description

200

Session successfully ended

PUT `/play/{mode}/{category}` Update the questions and the user-submitted answers in the db

A random question will come up from 1's to 12's in the selected category. The mode will be matched up along with the userid by reference to update the array of questions/answers the user has answered.

Parameters

Try it out

Name

Description

mode * required

string

(path)

The mode of the question: practice/evaluation

mode - The mode of the question: practice/evaluation

category * required

integer

(path)

The category of the question (i.e. 3's). If in 'challenge' mode, category not required. Note: a mode of 'challenge' was removed from MVP but placed as a stretch goal. We should note that if this stretch goal is implemented, a mode of 'challenge' is valid.

category - The category of the question (i.e. 3's). If in 'challen

body * required

object

(body)

GameData object

Example Value Model

```
{
  "userId": 123456789,
  "rowId": 123456,
  "problemsAndAnswers": [
    "3x6=10",
    "3x2=6",
    "3x1=3"
  ],
  "mode": "practice",
  "date": "03/18/2021 1:52 pm",
  "questionsAttempted": 10,
  "questionsCorrect": 7,
  "lengthOfTime": "4 min 40 sec"
}
```

Parameter content type

application/json

Responses

Response content type

application/xml

Code**Description**

201

Successful Operation - Data Created or Updated in DB

POST**/play/completed/** Saves all questions and answers for category just completed

Saves all questions and answers for category just completed

Parameters[Try it out](#)**Name****Description****body** * required

GameData object

object

(body)

Example Value Model

```
{
  "userId": 123456789,
  "rowId": 123456,
  "problemsAndAnswers": [
    "3x6=10",
    "3x2=6",
    "3x1=3"
  ],
  "mode": "practice",
  "date": "03/18/2021 1:52 pm",
  "questionsAttempted": 10,
  "questionsCorrect": 7,
  "lengthOfTime": "4 min 40 sec"
}
```

Parameter content type

[application/json](#)**Responses**

Response content type

[application/xml](#)**Code****Description**

Code

Description

200

HTTP OK - category questions and answers successfully retrieved

Example Value Model

```
<?xml version="1.0" encoding="UTF-8"?>
<GameData>
  <userId>123456789</userId>
  <rowId>123456</rowId>
  <problemsAndAnswers>3x6=10</problemsAndAnswers>
  <problemsAndAnswers>3x2=6</problemsAndAnswers>
  <problemsAndAnswers>3x1=3</problemsAndAnswers>
  <mode>practice</mode>
  <date>03/18/2021 1:52 pm</date>
  <questionsAttempted>10</questionsAttempted>
  <questionsCorrect>7</questionsCorrect>
  <lengthOfTime>4 min 40 sec</lengthOfTime>
</GameData>
```

400

Invalid category ID supplied

404

Game not found

history



GET

/history/{userId} Retrieves all history based on user

Parameters

Try it out

Name

Description

userId * required

UserId from current session

string

(path)

userId - UserId from current session

Responses

Response content type

application/xml

Code

Description

200

HTTP OK - history successfully retrieved

Example Value Model

```
<?xml version="1.0" encoding="UTF-8"?>
<GameData>
  <userId>123456789</userId>
  <rowId>123456</rowId>
  <problemsAndAnswers>3x6=10</problemsAndAnswers>
  <problemsAndAnswers>3x2=6</problemsAndAnswers>
  <problemsAndAnswers>3x1=3</problemsAndAnswers>
  <mode>practice</mode>
  <date>03/18/2021 1:52 pm</date>
  <questionsAttempted>10</questionsAttempted>
  <questionsCorrect>7</questionsCorrect>
  <lengthOfTime>4 min 40 sec</lengthOfTime>
</GameData>
```

400

Invalid userId supplied. Could not retrieve history

GET

/resultData/{rowId} Retrieves all history for a particular session based on row Id selected from UI

Parameters

Try it out

Name

Description

rowId * required

RowId selected from row in UI

string

(path)

rowId - RowId selected from row in UI

Responses

Response content type

application/xml

Code

Description

Code

Description

200

HTTP OK - result data successfully retrieved

Example Value Model

```
<?xml version="1.0" encoding="UTF-8"?>
<GameData>
  <userId>123456789</userId>
  <rowId>123456</rowId>
  <problemsAndAnswers>3x6=10</problemsAndAnswers>
  <problemsAndAnswers>3x2=6</problemsAndAnswers>
  <problemsAndAnswers>3x1=3</problemsAndAnswers>
  <mode>practice</mode>
  <date>03/18/2021 1:52 pm</date>
  <questionsAttempted>10</questionsAttempted>
  <questionsCorrect>7</questionsCorrect>
  <lengthOfTime>4 min 40 sec</lengthOfTime>
</GameData>
```

400

Invalid rowId supplied. Could not retrieve data

Models

```
GameData {
  userId          integer
  problemsAndAnswers [...]
  mode            string
  date            string
  questionsAttempted integer
  questionsCorrect integer
  lengthOfTime    string
}
```

```
example: OrderedMap { "userId": 123456789, "rowId": 123456, "problemsAndAnswers": List [
"3x6=10", "3x2=6", "3x1=3" ], "mode": "practice", "date": "03/18/2021 1:52 pm",
"questionsAttempted": 10, "questionsCorrect": 7, "lengthOfTime": "4 min 40 sec" }
```



UserData {

| | |
|------------------------|----------------------|
| <code>_id</code> | <code>string</code> |
| <code>firstName</code> | <code>string</code> |
| <code>lastName</code> | <code>string</code> |
| <code>isStudent</code> | <code>boolean</code> |
| <code>isParent</code> | <code>boolean</code> |
| <code>isTeacher</code> | <code>boolean</code> |

}

example: OrderedMap { "_id": "123456789", "firstName": "Test Student FN", "lastName": "Test Student LN", "isStudent": true, "isParent": false, "isTeacher": false }