Back-End Web Development

ST0503

Assignment 2

API Documentation

Alden Chia Yu Xiang

2205584

DISM/FT/2A/01

Table of Contents

[Basic Features 5](#_Toc142218910)

[Endpoint 1: POST /user/login 5](#_Toc142218911)

[Postman success response 5](#_Toc142218912)

[Postman error response 6](#_Toc142218913)

[Endpoint 2: PUT /user/signup 7](#_Toc142218914)

[Postman success response 8](#_Toc142218915)

[Postman error response 8](#_Toc142218916)

[Endpoint 3: PUT /user/updatePass 10](#_Toc142218917)

[Postman success response 11](#_Toc142218918)

[Postman error response 11](#_Toc142218919)

[Endpoint 4: GET /get/username 13](#_Toc142218920)

[Postman success response 13](#_Toc142218921)

[Postman error response 14](#_Toc142218922)

[Endpoint 5: GET /get/games 16](#_Toc142218923)

[Postman success response 16](#_Toc142218924)

[Postman error response 17](#_Toc142218925)

[Endpoint 6: GET /search/game 17](#_Toc142218926)

[Postman success response 18](#_Toc142218927)

[Postman error response 18](#_Toc142218928)

[Endpoint 7: GET /game/:gameid/:platformname 19](#_Toc142218929)

[Postman success response 19](#_Toc142218930)

[Postman error response 20](#_Toc142218931)

[Endpoint 8: GET /reviews 21](#_Toc142218932)

[Postman success response 21](#_Toc142218933)

[Postman error response 21](#_Toc142218934)

[Endpoint 9: POST /add/review 22](#_Toc142218935)

[Postman success response 23](#_Toc142218936)

[Postman error response 23](#_Toc142218937)

[Endpoint 10: GET /get/platforms 26](#_Toc142218938)

[Postman success response 26](#_Toc142218939)

[Postman error response 27](#_Toc142218940)

[Endpoint 11: POST /add/platform 27](#_Toc142218941)

[Postman success response 28](#_Toc142218942)

[Postman error response 28](#_Toc142218943)

[Endpoint 12: GET /get/categories 31](#_Toc142218944)

[Postman success response 31](#_Toc142218945)

[Postman error response 31](#_Toc142218946)

[Endpoint 13: POST /add/game 32](#_Toc142218947)

[Postman success response 32](#_Toc142218948)

[Postman error response 33](#_Toc142218949)

[Advance Features 36](#_Toc142218950)

[API 14: POST /add/image 36](#_Toc142218951)

[Postman success response 36](#_Toc142218952)

[Postman error response 37](#_Toc142218953)

[Endpoint 15: POST /add/cart 40](#_Toc142218954)

[Postman success response 40](#_Toc142218955)

[Postman error response 41](#_Toc142218956)

[Endpoint 16: DELETE /del/cart 43](#_Toc142218957)

[Postman success response 43](#_Toc142218958)

[Postman error response 44](#_Toc142218959)

[Endpoint 17: GET /check/cart 46](#_Toc142218960)

[Postman success response 46](#_Toc142218961)

[Postman error response 47](#_Toc142218962)

[Endpoint 18: GET /view/cart 48](#_Toc142218963)

[Postman success response 48](#_Toc142218964)

[Postman error response 49](#_Toc142218965)

[API 19: GET /user/games 50](#_Toc142218966)

[Postman success response 50](#_Toc142218967)

[Postman error response 51](#_Toc142218968)

[API 20: GET /owned/games 53](#_Toc142218969)

[Postman success response 53](#_Toc142218970)

[Postman error response 54](#_Toc142218971)

[API 21: POST /bought/games 55](#_Toc142218972)

[Postman success response 56](#_Toc142218973)

[Postman error response 56](#_Toc142218974)

[API 22: GET /verify/admin 59](#_Toc142218975)

[Postman success response 59](#_Toc142218976)

[Postman error response 59](#_Toc142218977)

[Tables 60](#_Toc142218978)

[Endpoints 60](#_Toc142218979)

[Error Messages 61](#_Toc142218980)

# Basic Features

## Endpoint 1: POST /user/login

Description: Match user credentials in database and webpage

Usage: For authenticating users in login page

Request parameters: N/A

Request body:

{

username: admin

password: admin

}

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

Wrong credentials: Bad request (400)

A screenshot of a computer

Description automatically generated

Missing fields: Bad request (400)

A screenshot of a computer

Description automatically generated

## Endpoint 2: PUT /user/signup

Description: Insert new row to table users

Usage: To add new user into the database in signup page

Request parameters: N/A

Request body:

{

username: Test

password: Test@123

email: Testing@gmail.com

member\_type: Gold (e.g. Bronze, Silver, Diamond)

points: 0

profile\_picture\_url: https://testing.org/testing\_image

}

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

Duplicate Entry (422)

A screenshot of a computer

Description automatically generated

Missing field: Bad Request (400)

A screenshot of a computer

Description automatically generated

Invalid memberType: Bad Request (400)

A screenshot of a computer

Description automatically generated

## Endpoint 3: PUT /user/updatePass

Description: Update password column in table users

Usage: Allow users to change their password in changePass page

Request parameters: N/A

Request body:

{

newPassword: Password

oldPassword: P@sSw0%d

email: user@gmail.com

}

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

Invalid email: Bad Request (400)

A screenshot of a computer

Description automatically generated

Missing field: Bad Request (400)

A screenshot of a computer

Description automatically generated

## Endpoint 4: GET /get/username

**verifyToken:** Verify if user have token (login)

Description: Retrieve username from users table

Usage: To display username in home page

Request parameters: N/A

Request body: N/A

Request query: userid (e.g. 1, 3, 7)

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

Invalid body: Bad Request (400)

A screenshot of a computer

Description automatically generated

Missing field: Bad Request (400)

A screenshot of a computer

Description automatically generated

No token: Forbidden (403)

A screenshot of a computer

Description automatically generated

## Endpoint 5: GET /get/games

Description: Retrieves game details from games table

Usage: To display information in games page

Request parameters: N/A

Request body: N/A

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

## Endpoint 6: GET /search/game

Description: Retrieves game details from games table

Usage: To display games information in dynamic games page

Request parameters: N/A

Request body: N/A

Request query: title (e.g. Valorant, League), platformname (e.g. PC, PlayStation)

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error

A screenshot of a computer

Description automatically generated

Missing field: Bad Request (400)

A screenshot of a computer

Description automatically generated

## Endpoint 7: GET /game/:gameid/:platformname

Description: Retrieves games and platform information

Usage: To display games and platform information in individual games page

Request parameters: gameid (e.g. 1, 3, 7), platformname(e.g. Xbox, play station 4)

Request body: N/A

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

gameid does not exist with platformname/invalid gameid or platformname: Not found (404)

A screenshot of a computer

Description automatically generated

## Endpoint 8: GET /reviews

Description: Retrieves reviews data from review table

Usage: To display reviews for game in individual game page

Request parameters: N/A

Request body: N/A

Request query: gameid (e.g. 1, 2, 6)

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

Missing field: Bad Request (400)

A screenshot of a computer

Description automatically generated

## Endpoint 9: POST /add/review

**verifyToken:** Verify if user have token (login)

Description: Insert into review table

Usage: Allow user to add their own review for games

Request parameters: N/A

Request body:

{

userid: 3

gameid: 1

content: Best game ever!

rating : 5

}

### Postman success response

A screenshot of a computer

Description automatically generated



### Postman error response

Invalid parameter: Not Found (404)

A screenshot of a computer

Description automatically generated

Missing field: Bad Request (400)

A screenshot of a computer

Description automatically generated

Invalid rating (rating < 0 or rating > 5): Bad Request (400)

A screenshot of a computer

Description automatically generated

gameid or userid not found: Not Found (403)

A screenshot of a computer

Description automatically generated

No token: Forbidden (403)

A screenshot of a computer

Description automatically generated

## Endpoint 10: GET /get/platforms

Description: Retrieve data from platform table

Usage: To be displayed in dropdown options

Request parameters: N/A

Request body: N/A

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

## Endpoint 11: POST /add/platform

**verifyToken**: Verify if user have token (login)

**verifyAdmin**: Verify if user is admin

Description: Insert data into platform table

Usage: For admin users to insert new platform

Request parameters: N/A

Request body:

{

platformname: Smart Fridge

description: Best gaming device ever release!

}

### Postman success response

A screenshot of a computer

Description automatically generated



### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

Missing field: Bad Request (400)

A screenshot of a computer

Description automatically generated

Duplicate Entry (422)

A screenshot of a computer

Description automatically generated

No token: Forbidden (403)

A screenshot of a computer

Description automatically generated

Unauthorised user: Forbidden (403)

A screenshot of a computer

Description automatically generated

## Endpoint 12: GET /get/categories

Description: Retrieve data from category table

Usage: To display categories for admin to choose when admin wants to add game

Request parameters: N/A

Request body: N/A

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

## Endpoint 13: POST /add/game

**verifyToken**: Verify if user have token (login)

**verifyAdmin**: Verify if user is admin

Description: Insert data into games table

Usage: For admin users to insert new game

Request parameters: N/A

Request body:

{

title: newGame

description: gameDescription

platformid: 1, 4

price: 33, 30

categoryid: 1, 7, 13

year: 2022

}

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

Missing field: Bad Request (400)

A screenshot of a computer

Description automatically generated

Length of price != points != platformid: Bad Request (400)

A screenshot of a computer

Description automatically generated

Duplicate Entry (422)

A screenshot of a computer

Description automatically generated

No token: Forbidden (403)

A screenshot of a computer

Description automatically generated

Unauthorised user: Forbidden (403)

A screenshot of a computer

Description automatically generated

# Advance Features

## API 14: POST /add/image

Description: Adding an image into a folder and updating gamesimage table

Usage: Enable admin to add image for game

Request parameters:

Request body:

{

title: newGame

}

Request files: gameImage.jpeg

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

Duplicate Entry (422)

A screenshot of a computer

Description automatically generated

Invalid gameid: Not Found (404)

A screenshot of a computer

Description automatically generated

Missing field: Not Found (404)

A screenshot of a computer

Description automatically generated

Invalid file type: Unsupported Media Type (415)

A screenshot of a computer

Description automatically generated

Payload Too Large (413)

A screenshot of a computer

Description automatically generated

## Endpoint 15: POST /add/cart

Description: Insert data into cart table

Usage: When user click ‘Buy Now’, the details will be added to cart

Request parameters: N/A

Request body:

{

userid: 13

title: Valorant

platformname: PC

}

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error

A screenshot of a computer

Description automatically generated

Duplicate Entry (422)

A screenshot of a computer

Description automatically generated

Missing field: Bad request (400)

A screenshot of a computer

Description automatically generated

gameid does not exist in platform: Not Found (404)

A screenshot of a computer

Description automatically generated

## Endpoint 16: DELETE /del/cart

Description: Remove data from cart table

Usage: When user click ‘Remove’, the details will be deleted from cart

Request parameters: N/A

Request body:

{

userid: 13

title: Valorant

platformname: PC

}

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

Missing field: Bad Request (400)

A screenshot of a computer

Description automatically generated

Row does not exist: Not Found (404)

A screenshot of a computer

Description automatically generated

title or platformname does not exist: Not Found (404)

A screenshot of a computer

Description automatically generated

## Endpoint 17: GET /check/cart

Description: Retrieve information from cart table

Usage: To check if game is added in user cart

Request parameters: N/A

Request body: N/A

Request query: userid (e.g. 1, 3, 7), title (e.g. Valorant, League), platformname (e.g. PC, PlayStation)

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

Row does not exist: Bad Request (400)

A screenshot of a computer

Description automatically generated

Missing field: Bad Request (400)

A screenshot of a computer

Description automatically generated

## Endpoint 18: GET /view/cart

Description: Retrieve data from cart table

Usage: To display data in checkout and cart page for user to see.

Request parameters: N/A

Request body: N/A

Request query: userid (e.g. 1, 3, 7)

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

Missing field: Bad Request (400)

A screenshot of a computer

Description automatically generated

## API 19: GET /user/games

Description: Retrieve data from usergames table

Usage: To determine if user already own games

Request parameters: N/A

Request body: N/A

Request query: userid (e.g. 1, 3, 7), title (e.g. Valorant, League), platformname (e.g. PC, PlayStation)

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)A screenshot of a computer

Description automatically generated

Missing field: Bad Request (400)

A screenshot of a computer

Description automatically generated

game does not exist in platform: Not Found (404)

A screenshot of a computer

Description automatically generated

Row does not exist: Not Found (404)

A screenshot of a computer

Description automatically generated

Invalid userid: Not Found (404)

A screenshot of a computer

Description automatically generated

## API 20: GET /owned/games

Description: Retrieve data from usergames table

Usage: To show users the game they own in profile page

Request parameters: N/A

Request body: N/A

Request query: userid (e.g. 1, 3, 7)

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

Missing field: Bad Request (400)

A screenshot of a computer

Description automatically generated

userid does not exist: Not Found (404)

A screenshot of a computer

Description automatically generated

## API 21: POST /bought/games

Description: Insert data from usergames table

Usage: Allow user to buy games

Request parameters: N/A

Request body:

{

userid: 13

title: Valorant, Overwatch

platformname: PC, PC

}

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Internal Server Error (500)

A screenshot of a computer

Description automatically generated

game does not exist in platform: Not Found (404)

A screenshot of a computer

Description automatically generated

Missing field: Bad Request (400)

A screenshot of a computer

Description automatically generated

title or platformname does not exist: Not Found (404)

A screenshot of a computer

Description automatically generated

Duplicate Entry (422)

A screenshot of a computer

Description automatically generated

## API 22: GET /verify/admin

**verifyToken**: Verify if user have token (login)

**verifyAdmin**: Verify if user is admin

Description: Check user’s token if they are admin

Usage: Check user’s token if they are admin

### Postman success response

A screenshot of a computer

Description automatically generated

### Postman error response

Unauthorised user: Forbidden (403)

A screenshot of a computer

Description automatically generated

# Tables

## Endpoints

|  |  |  |  |
| --- | --- | --- | --- |
| Endpoint | Method | verifyToken? | verifyAdmin? |
| /user/login | POST | No | No |
| /user/signup | PUT | No | No |
| /user/updatePass | GET | No | No |
| /get/username | GET | Yes | No |
| /get/games | GET | No | No |
| /search/game | GET | No | No |
| /game/:gameid/:platformname | GET | No | No |
| /reviews | GET | No | No |
| /add/review | POST | Yes | No |
| /get/platforms | GET | No | No |
| /add/platform | POST | Yes | Yes |
| /get/categories | GET | No | No |
| /add/game | POST | Yes | Yes |
| /add/image | POST | No | No |
| /add/cart | POST | No | No |
| /del/cart | DELETE | No | No |
| /check/cart | GET | No | No |
| /view/cart | GET | No | No |
| /user/games | GET | No | No |
| /owned/games | GET | No | No |
| /bought/games | POST | No | No |
| /verify/admin | GET | Yes | Yes |

## Error Messages

|  |  |
| --- | --- |
| Error Response Code | Return |
| 500 | Internal Server Error |
| 422 | Un-processable Content |
| 404 | Not Found |
| 400 | Bad request |
| 405 | Method Not Allowed |
| 413 | Payload too large |
| 415 | Unsupported media type |

-END-