

Service Layers

Alexei Cogdill

Overview

The backend service for my web application will be built using the Flask web framework. The use of simple REST API will be utilized on the frontend to call which will be used to create and manipulate data.

Specification:

User Route:

1. Get user scores:

Method: GET

URL: <https://memory-coreplex.herokuapp.com/api/profile/stats/:id>

Purpose: After the user has submitted scores for the game, they will get posted on the leaderboard if they meet the right criteria (making top 10, 15, etc). This will be used so that when players navigate to their profile, it will pull the correct user stats based on the id for that user and display the scores. This will allow the user to know what their top scores are to beat.

Example request:

Curl-request GET -url

<https://memory-coreplex.herokuapp.com/api/profile/stats/:id>

Example response:

```
{
  "scoreData": [
    {
      "user_ID": "48"
      "display_name": "MemoryMaster"
    }
  ]
}
```

```

    "score": {
      "score": "293"
    },
    "leaderboard": {
      "difficulty": "Medium"
      "game": "Game 1"
    }
  }
]
}

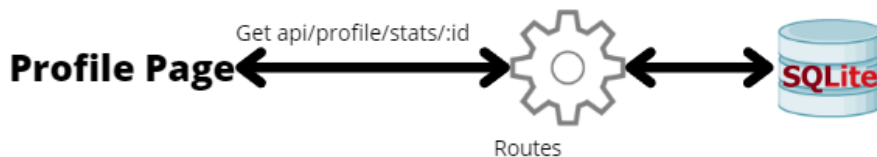
```

Example error response:

```

{
  "status": 400,
  "message": "User scores could not be retrieved at this time."
}

```



2. Get user info:

Method: GET

URL: <https://memory-coreplex.herokuapp.com/api/profile>

Purpose: When a person creates a user and navigates to their profile, this endpoint will be called from the front end and the data returned will be used to populate the profile page.

Example request:

Curl-request GET -url

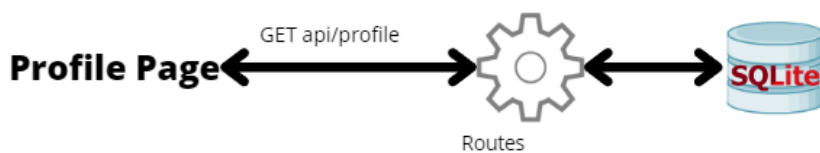
<https://memory-coreplex.herokuapp.com/api/profile>

Example response:

```
{
  "userData": [
    {
      "user_ID": "48"
      "username": "atkc"
      "display_name": "MemoryMaster"
      "name": "Alexei"
      "email": "alex@gmail.com"
    }
  ]
}
```

Example error response:

```
{
  "status": 400,
  "message": "User info cannot be found"
}
```



3. User/Admin Login

Method: GET

URL: <https://memory-coreplex.herokuapp.com/api/users/:username>

Purpose: When a person creates a user to access the games and leaderboard, or even potentially an admin of the site, once they click submit this endpoint will be called. The user will be redirected to the home page upon successful validation.

Example request:

Curl-request GET -url

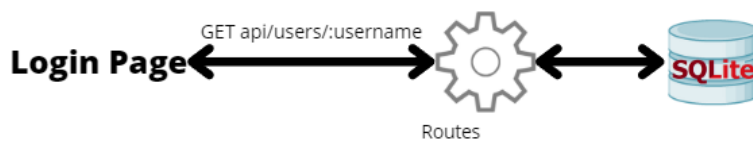
`https://memory-coreplex.herokuapp.com/api/profile/:username`

Example response:

```
{
  "data": [
    {
      "user_ID": "48"
      "name": "atkc"
      "display_name": "MemoryMaster"
      "email": "alex@gmail.com"
      "username": "alex123"
      "password_hash": "241830rh3hr329r32"
      "isAdmin": False
    }
  ]
}
```

Example error response:

```
{
  "status": 400,
  "message": "user record does not exist"
}
```



Review Route:

4. Create new review record:

Method: POST

URL: <https://memory-coreplex.herokuapp.com/api/review/newReview>

Purpose: When a registered user has experienced the memory game(s) offered, they can go to the review page and submit a review by submitting a form. After they fill out the form and hit submit, this endpoint will be called containing the form data. This data will be to the review approval page that site admin's see.

Example request:

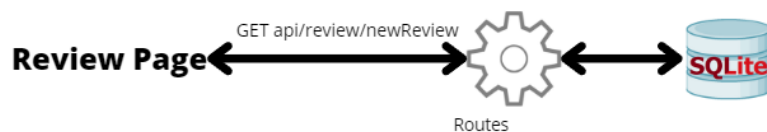
```
curl -request POST \
  --url https://memory-coreplex.herokuapp.com/api/review/newReview \
  --header 'Content-Type: application/json' \
  --data '{
    "title": "Good memory games, helped me improve"
    "message": "This memory game..... etc etc"
    "user": {
      "display_name": "MemoryMaster"
    },
  }
```

Success response:

```
{
  "title": "Good memory games, helped me improve"
  "message": "This memory game..... etc etc"
  "user": {
    "display_name": "MemoryMaster"
  },
  "review_ID": "2532523"
}
```

Error response:

```
[
  {
    "message": "\"title\" is required",
    "path": [
      "title"
    ],
    "type": "any.required",
    "context": {
      "key": "title",
      "label": "title"
    }
  }
]
```



5. Get Reviews:

Method: GET

URL: <https://memory-coreplex.herokuapp.com/api/reviews?isPending=<bool>>

Purpose: When a user clicks on the review's navigation item, this endpoint will be called. The user will be redirected to the reviews page where the data from the endpoint will be returned, loading all the reviews on the page

Example requests:

Curl -request GET -url

<https://memory-coreplex.herokuapp.com/api/reviews?isPending=<bool>>

Success Response:

```
{
  "data": [
```

```

    { <3> },
    { <3> },
    { <3> },
    {
      "review_ID": "214812y180hr"
      "display_name": "MemoryMaster"
      "title": "Good Game"
      "message": "This site is good"
      "isPending": false
    }
  ]
}

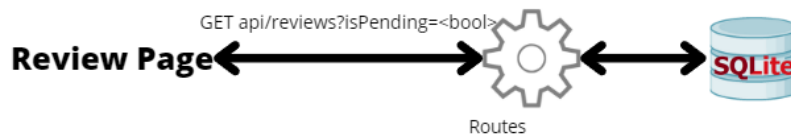
```

Error response:

```

{
  "status": 400,
  "message": "Could not retrieve reviews."
}

```



6. Get pending reviews:

Method: GET

URL: <https://memory-coreplex.herokuapp.com/api/reviews?isPending=<bool>>

Purpose: When an admin is logged in and click the Reviews Approval menu button, this endpoint will be called. They will then be redirected to the reviews approval page where the data will render from the endpoint.

Example requests:

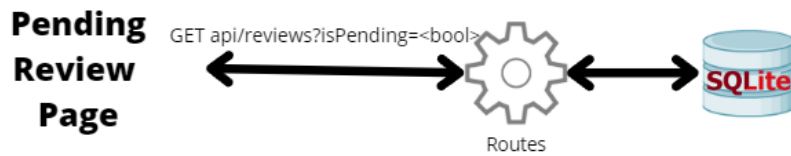
```
Curl -request GET https://memory-  
coreplex.herokuapp.com/api/reviews?isPending=<bool>
```

Success Response:

```
{  
  "data": [  
    {  
      "review_ID": "1221532532",  
      "display_name": "MemoryMaster"  
      "title": "Good Game"  
      "message": "This site is good"  
      "isPending": false  
    },  
    { <3> },  
    { <3> },  
    { <3> },  
  ]  
}
```

Error Response:

```
{  
  "status": 400,  
  "message": "Could not retrieve reviews."  
}
```

7. Approve Review:

Method: PUT

URL: <https://memory-coreplex.herokuapp.com/api/reviews/pending/approve/:id>

Purpose: When an admin clicks the submit button for the review, this endpoint will be called. The admin user will then be directed to the pending reviews page.

Example requests:

Curl -request PUT -url

<https://memory-coreplex.herokuapp.com/api/reviews/pending/approve/:id>

Success response:

```
{
  "message": "The review from "/display_name/" has been approved."
}
```

Error response:

```
{
  "status": 400,
  "message": "There was an error approving the review. Please try again."
}
```



8. Deny Review:

Method: PUT

URL: <https://memory-coreplex.herokuapp.com/api/reviews/pending/deny/:id>

Purpose: When an admin user clicks the reject button on a pending review, this endpoint will be called. Once submitted and processed, the user will be redirected to the pending reviews page to view other reviews.

Example requests:

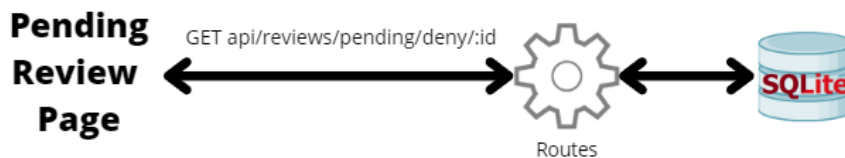
```
Curl -request PUT -url : https://memory-coreplex.herokuapp.com/api/reviews/pending/deny/:id
```

Success response:

```
{  
  "message": "Review has been denied... removing from list."  
}
```

Error response:

```
{  
  "status": 400,  
  "message": "There was an error rejecting review. Please try again."  
}
```



Score Route:

9. Create a new score record:

Method: POST

URL: <https://memory-coreplex.herokuapp.com/api/user/newScore>

Purpose: After the user has completed the game, they will be redirected to a page to submit their score. After they enter their initials and hit submit, this endpoint will be called with the user score data as the payload for the request. Once processed, the user will be redirected to the leaderboard page which will include the data returned from the endpoint.

Example request:

```
Curl -request POST \  
--url https://memory-coreplex.herokuapp.com/api/user/newScore \  
--header 'Content-Type:application/json' \  
--data '{  
    "display_name": "ATKC",  
    "score: {  
        "score": 82,  
        "time": 120  
    }'  
'
```

Example response:

```
{  
  "message": "User has submitted score."  
}
```

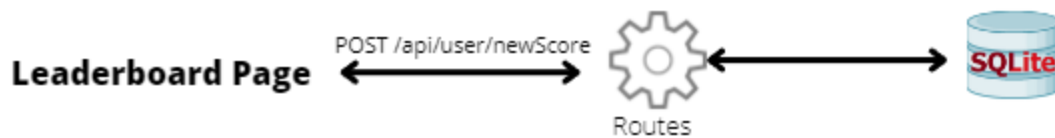
Example error response:

```

[
  {
    "message": "\"userinitials\" are required",
    "path": [
      "userinitials"
    ],
    "type": "any.required",
    "context": {
      "key": "userinitials",
      "label": "userinitials"
    }
  }
]

```

Diagram:



Leaderboard Route:

10. Get game scores by difficulty:

Method: GET

URL: <https://memory-coreplex.herokuapp.com/api/leaderboard/:difficulty>

Purpose: When a user clicks which game's difficulty, they'd like to display the high scores for, the data returned from the endpoint will be rendered in the leaderboard page.

Example request:

Curl `-request GET -url`

`https://memory-coreplex.herokuapp.com/api/leaderboard/:difficulty`

Example response:

```
{
```

```

“data”: [
  { <- 5 -> },
  { <- 5 -> },
  { <- 5 -> },
  { <- 5 -> },
  { <- 5 -> },
  { <- 5 -> },
  { <- 5 -> },
  {
    “id”: “435353125f32fsf2”,
    “difficulty”: “medium”,
    “game”: “game1”,
    “user”: {
      “userinitials”: “ATKC”
    },
    “score”: {
      “score”: 23,
      “time”: 10
    }
  }
]

```

Example error response:

```

{
  “status”: 400,
  “message”: “Medium level leaderboard scores could not be retrieved.”
}

```

Diagram:



Games Route:

11. Get games

Method: GET

URL: <https://memory-coreplex.herokuapp.com/api/games>

Example requests:

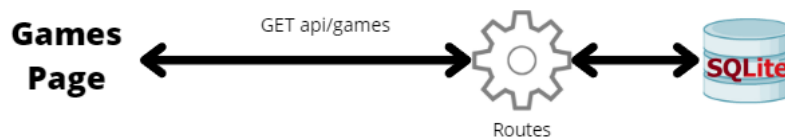
Curl -request GET <https://memory-coreplex.herokuapp.com/api/games>

Success Response:

```
{  
  "message": "Games successfully loaded"  
}
```

Error Response:

```
{  
  "message": "Game did not load, please refresh the page"  
}
```



Game Submission Route: (Stretch Feature)

1. Get pending game submissions

Method: GET

URL: <https://memory-coreplex.herokuapp.com/api/gamesubmissions>

Purpose: When an admin is logged in and click the Game Submissions menu button, this endpoint will be called. They will then be redirected to the game submissions page where the data will render from the endpoint.

Example requests:

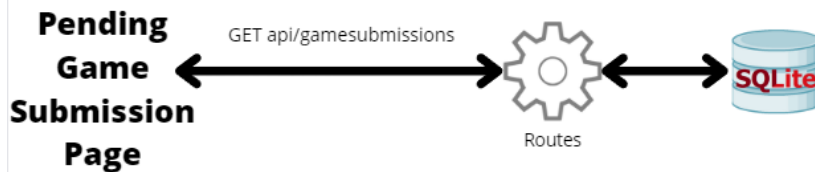
Curl -request GET <https://memory-coreplex.herokuapp.com/api/gamesubmissions>

Success Response:

```
{
  "data": [
    {
      "game_submission_ID": "1221532532",
      "name": "David Smith"
      "email": "test@gmail.com"
      "game_name": "Guess the Number"
      "game_link": "https://gamelink.com
      "message": "I believe my game should be implemented on your platform
because of ...."
    },
    { <3> },
    { <3> },
    { <3> },
  ]
}
```

Error Response:

```
{
  "status": 400,
  "message": "Could not retrieve game submissions."
}
```



2. Approve Game Submission:

Method: PUT

URL: <https://memory-coreplex.herokuapp.com/api/gamesubmission/pending/approve/:id>

Purpose: When an admin clicks the submit button for the submitted game, this endpoint will be called. The admin user will then be directed to the game submission page.

Example requests:

Curl -request PUT -url

<https://memory-coreplex.herokuapp.com/api/gamesubmission/pending/approve/:id>

Success response:

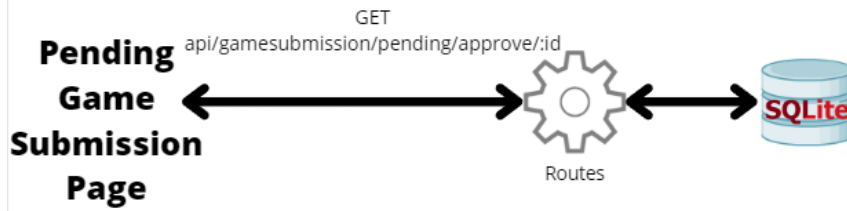
```
{
  "message": "The game, "/game_name/", has been approved. Email will be
sent to accepted user."
}
```

Error response:

```
{
  "status": 400,
```


“message”: “There was an error approving the game. Please try again.”

}



3. Deny Review:

Method: PUT

URL: <https://memory-coreplex.herokuapp.com/api/gamesubmission/pending/deny/:id>

Purpose: When an admin user clicks the reject button on a pending review, this endpoint will be called. Once submitted and processed, the user will be redirected to the pending reviews page to view other reviews.

Example requests:

Curl `-request PUT -url : https://memory-coreplex.herokuapp.com/api/gamesubmission/pending/deny/:id`

Success response:

{

“message”: “Game has been denied... removing from list.”

}

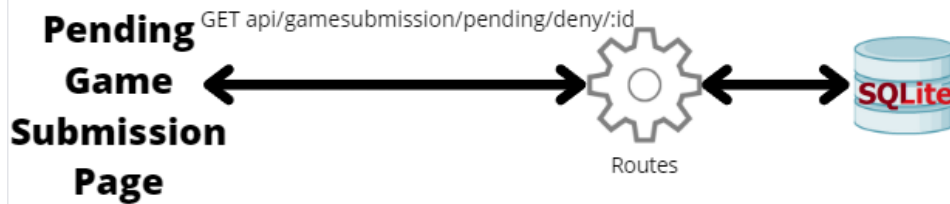
Error response:

{

“status”: 400,

“message”: “There was an error rejecting the game. Please try again.”

}



12. Create new game submission record:

Method: POST

URL: <https://memory-coreplex.herokuapp.com/api/gamesubmission/newGame>

Purpose: When someone fills out the form for requesting their game to be on the platform, this endpoint will be called. The data will be rendered whenever a site admin navigates to the game submission page.

Example request:

```
curl -request POST \  
  --url https://memory-coreplex.herokuapp.com/api/review/newReview \  
  --header 'Content-Type: application/json' \  
  --data '{  
    "name": "David Smith"  
    "email": "test@gmail.com"  
    "game_name": "Memory Tester"  
    "game_link": "gamelinkhere.com"  
    "message": "My game should be here because of...."  
  }'
```

Success response:

```
{  
  "name": "David Smith"
```

```
    "email": "test@gmail.com"
    "game_name": "Memory Tester"
    "game_link": "gamelinkhere.com"
    "message": "My game should be here because of...."
  },
  "gamesubmission_ID": "532532"
}
```

Error response:

```
[
  {
    "message": "\"game_link\" is required",
    "path": [
      "title"
    ],
    "type": "any.required",
    "context": {
      "key": "game_link",
      "label": "game_link"
    }
  }
]
```

