# CPSC-449-Web Backend Engineering: Project-2

# Wordle: RESTful Service

## Users Microservice:

### 1. Registering a new user (username, password)

The API is for registering a new user. If all the parameters are "NOT NULL" and passed successfully, we get a success message with status code 200.

**Endpoint**: /register
**Request:** http://tuffix-vm/register
**Method:** POST
**Parameters:** username, password

Sample JSON Request:
```
{
        "username": "sravani",
        "password": "sravani@22"
}
```
Success Response:
```
{
        "message": "Successfully registered!",
        "statusCode": 200
}
```

### 2. Authenticating a user (username, password)

This API authentication is used for user login. If the username and password match the values in the database, it returns a 200 status code and an "authenticated: true" response. If the entered user is "not found" or "wrong" then it returns an error status code of 401, with the message "Incorrect username or password".

**Endpoint:** /auth
**Method:** GET
**Parameters:** username, password
**Authorization:** Basic Auth

Success Response:
```
{
        "statusCode":200,
        "authenticated": "true"
}
```

## Game Microservice:

### 3. Registering a new game for already existing authenticated user

This API is for creating a new game for a specific user. The client will need to include the user id in the request body. If successful (user exists), the server will send a response that includes the id of the newly created game.
If the username is not authenticated it will result in status code 401 and a message to provide the username.

**Endpoint**: /game
**Method:** POST
**Request:** http://tuffix-vm/game
**Parameters:** none
**Authentication:** Basic auth

Success Response (200):
```
{
        "gameId": 18c42887-7c55-4e5f-916e-a3db5c111b53
        "guesses": 6
}
```

### 4. Retrieve game state

This API is for retrieving the game state of a specific game. The client will need to provide the game id in the URL. If successful (game exists), the server will send back a response that includes the state of the game.

**Endpoint:** /game/:gameId
**Method:** GET
**Request:** http://tuffix-vm/game/18c42887-7c55-4e5f-916e-a3db5c111b53
**Parameters:** gameId
**Authentication:** Basic auth

Success Response (200):
```
{
        "finished": false,
        "guessLeft": 6,
        "guessedWords":[ ]
}
```

## 5. List games for a user

This API is for returning the past games played by the user.

**Endpoint:** /my-games
**Method:** GET
**Request**: http://tuffix-vm/my-games
**Authentication:** Basic auth

Success Response(200):
```
{
        "Finished": false,
        "gameId": 18c42887-7c55-4e5f-916e-a3db5c111b53,
        "guessesLeft": 6
}
```