

Design Document and Guide for Backend RESTful API for XO (Tic-Tac-Toe)

Prepared by

Chollasak Anuwareepong

Junior Computer Engineering student of

King Mongkut's Institute of Technology Ladkrabang

This document will show my assignment design for
Backend internship position at ACUITMESH CO., LTD.

List of contents

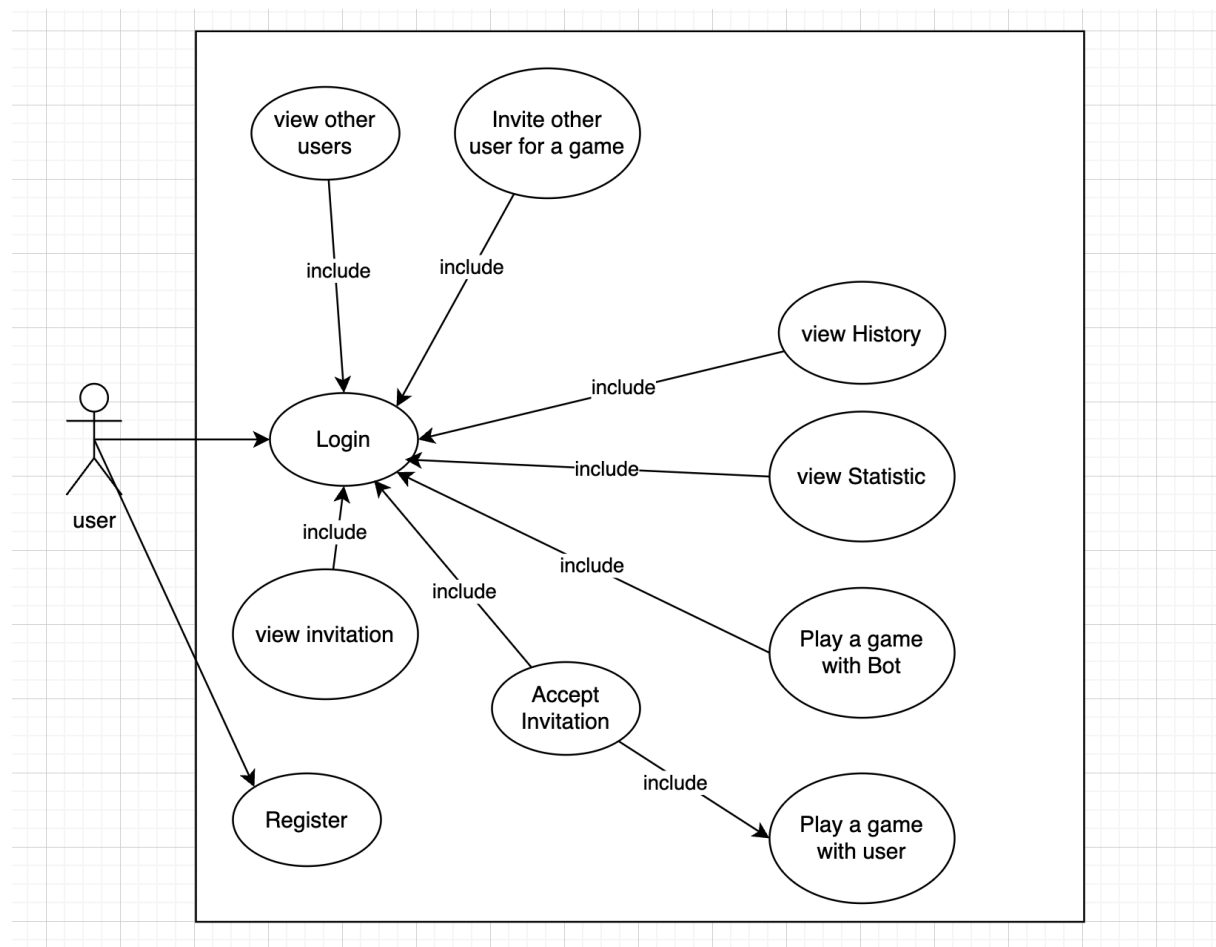
1. Tech Stack
2. Use Case Diagram
3. ER Diagram
4. Instruction guide

Tech Stack

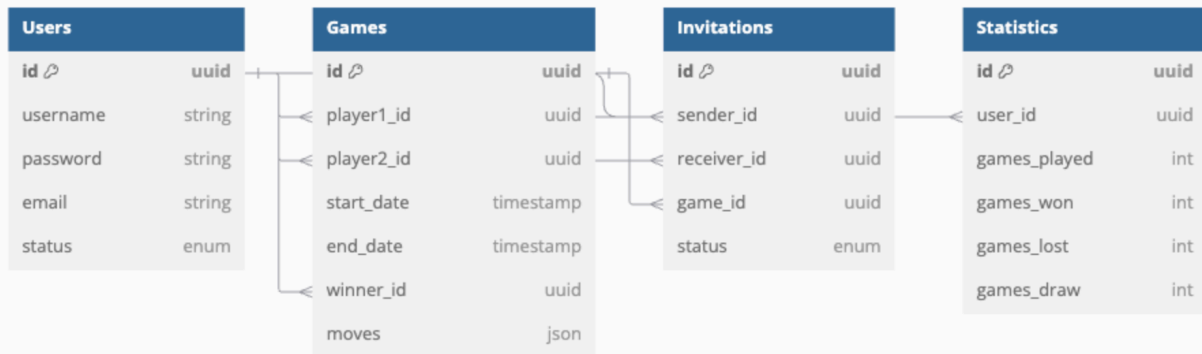
Backend Framework: **NestJS**

Database: **MongoDB**

Use Case Diagram

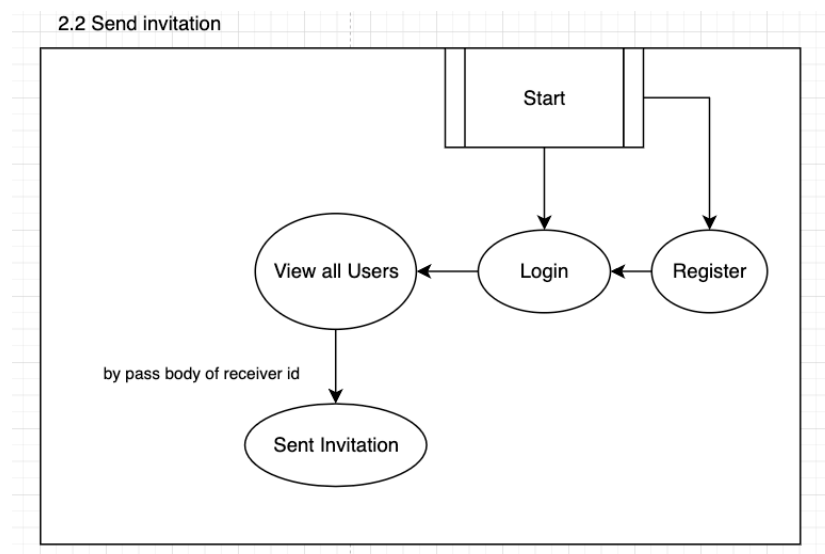
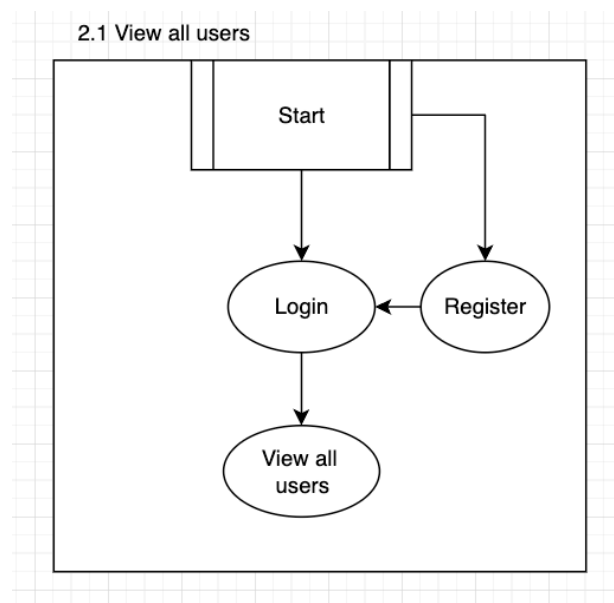


ER Diagram

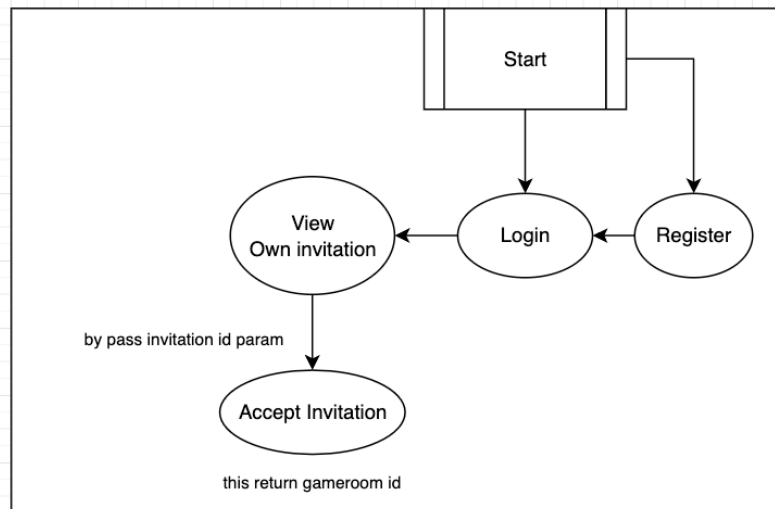


Instruction guide

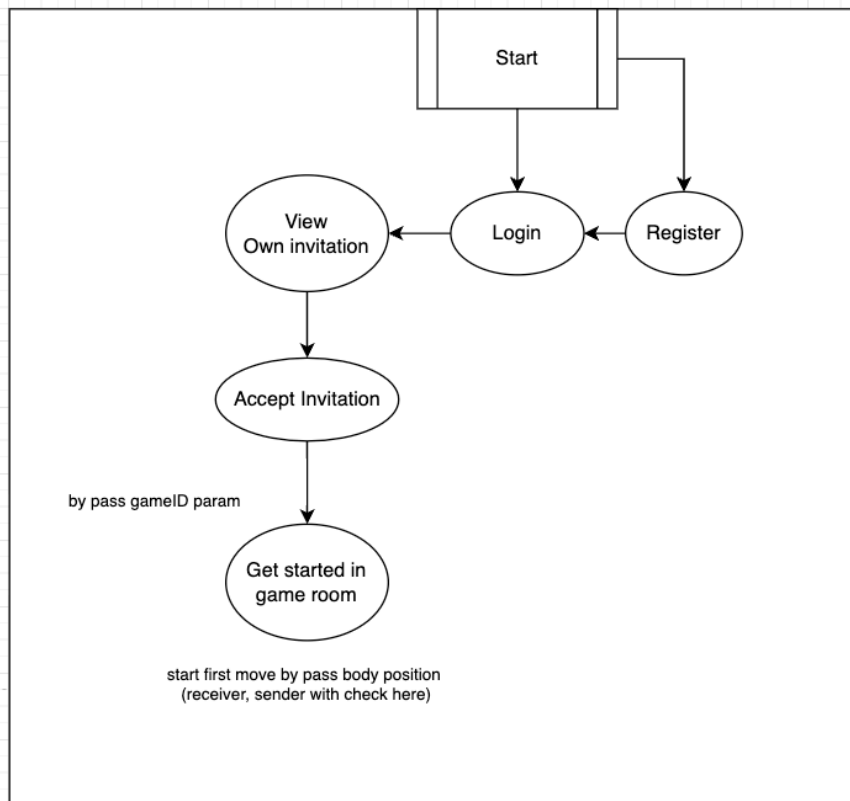
- Happy Flow



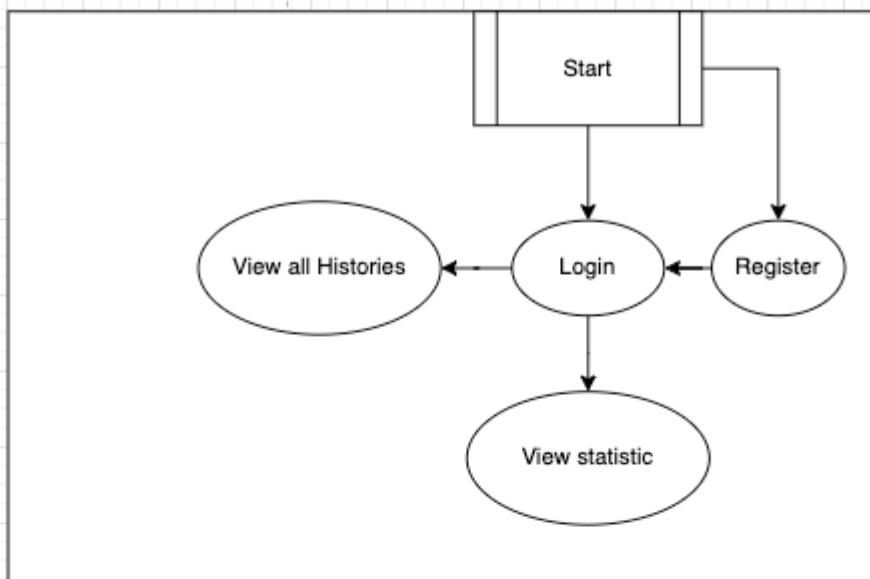
2.3 View own Invitation and accept



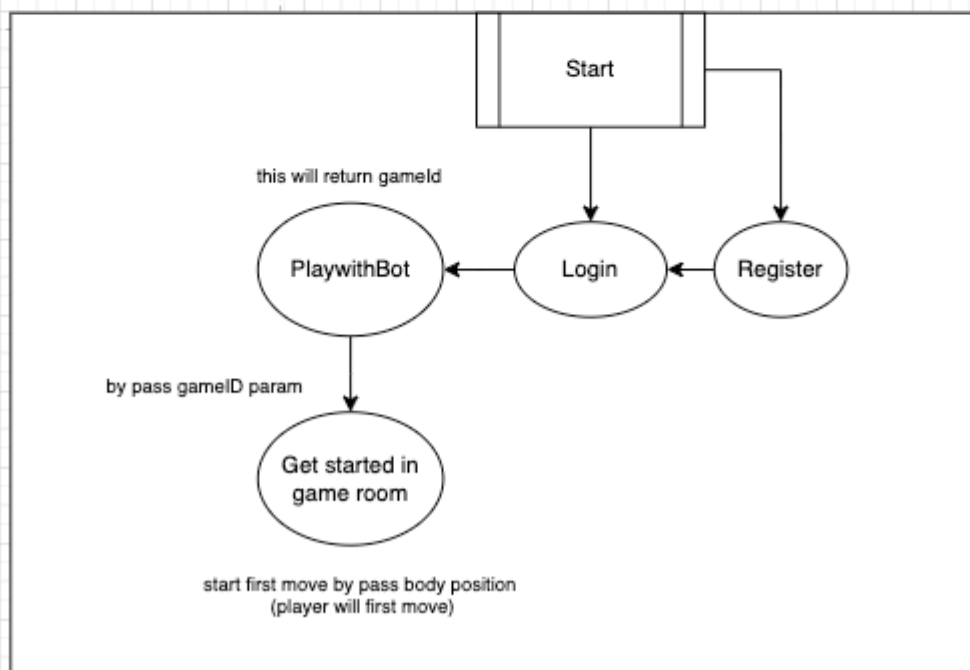
2.4 & 2.5 Start Playing



2.6 View own history and statistic

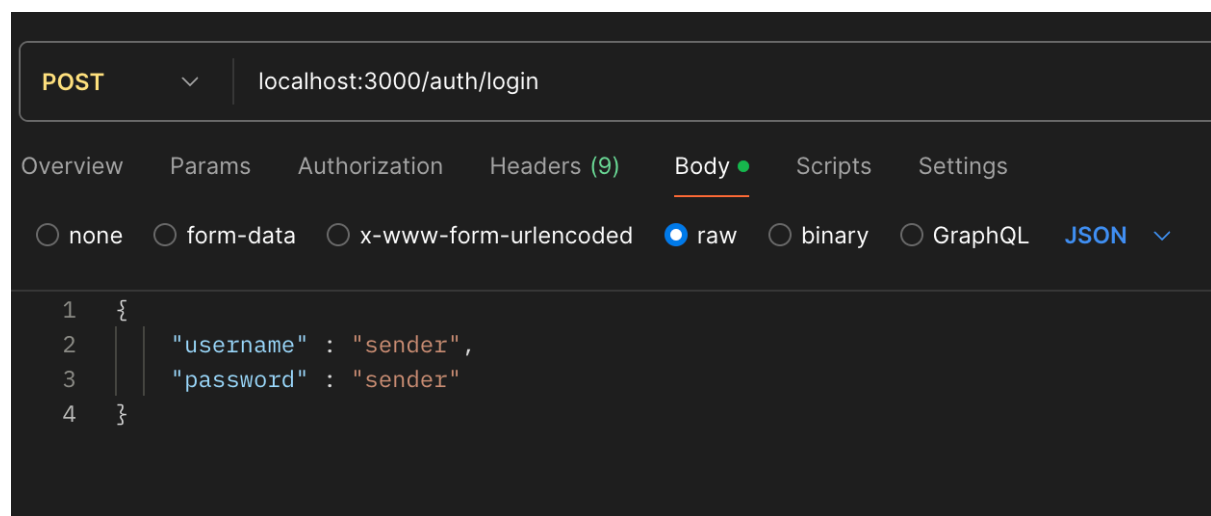
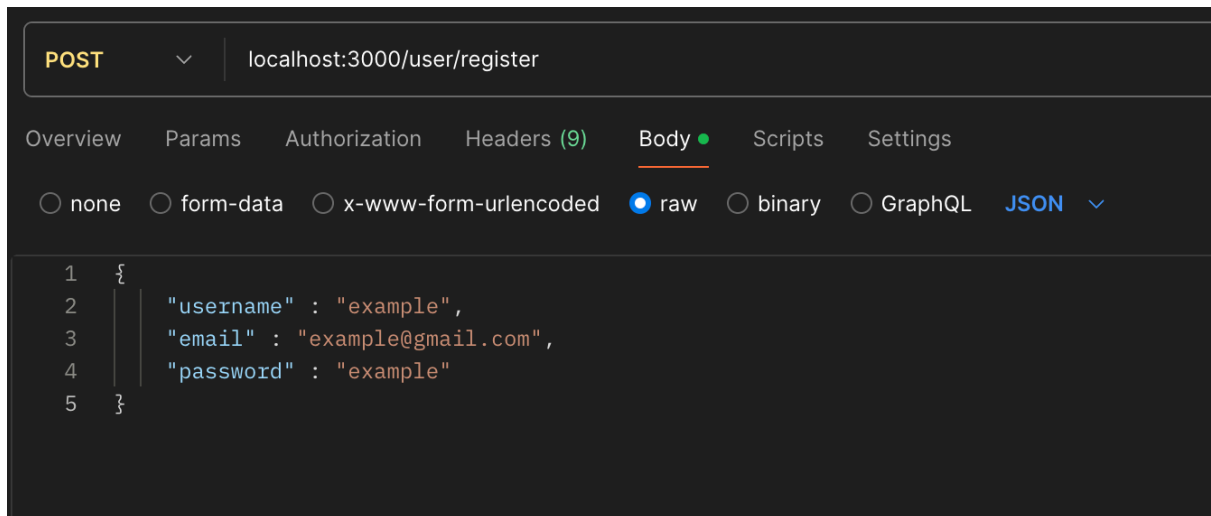


Extra Point

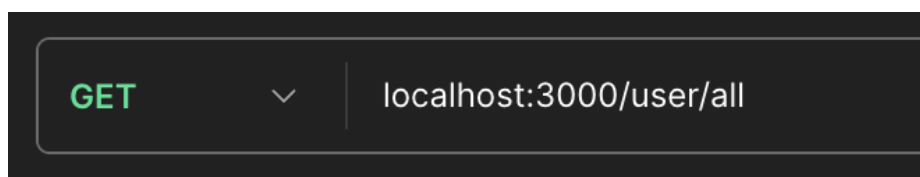


Postman Guide

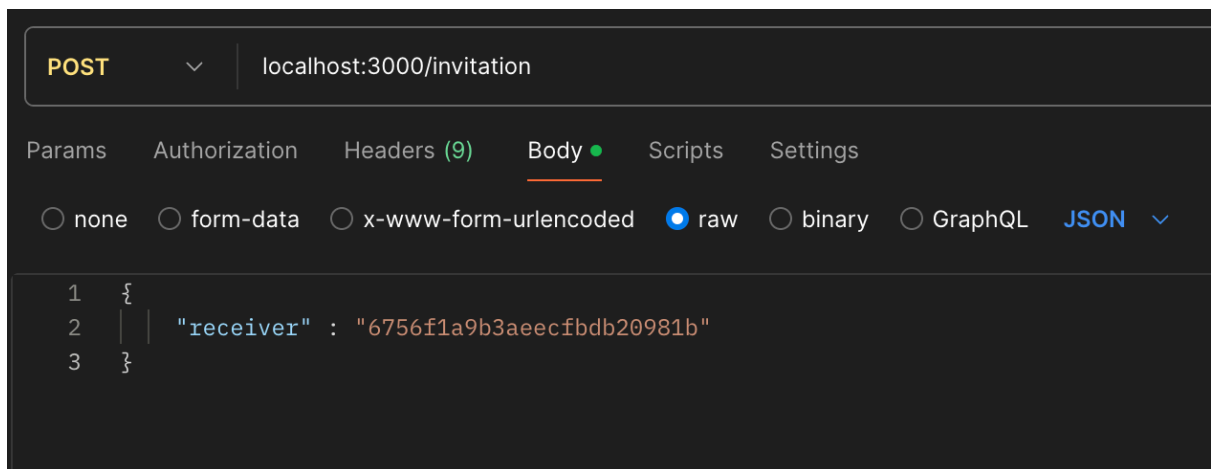
1. Register & Login



2. 1 View all users (after login)



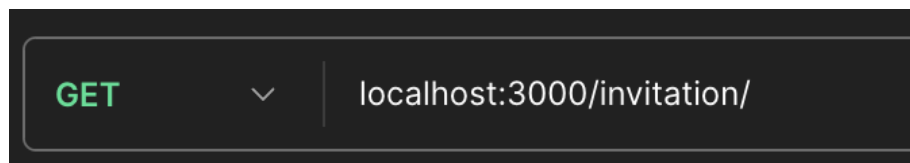
2.2 Send Invitation (Use id of user from 2.1 view all users)



A screenshot of a REST client interface. The top bar shows a **POST** method and the URL `localhost:3000/invitation`. Below the bar, there are tabs for **Params**, **Authorization**, **Headers (9)**, **Body** (which is selected and underlined), **Scripts**, and **Settings**. Under the **Body** tab, there are radio buttons for **none**, **form-data**, **x-www-form-urlencoded**, **raw** (which is selected), **binary**, and **GraphQL**. To the right of these is a dropdown menu showing **JSON**. The main area displays a JSON body with the following content:

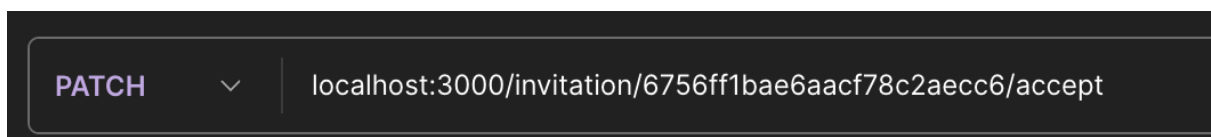
```
1 {  
2   | | "receiver" : "6756f1a9b3aeecfbdb20981b"  
3   }  
4 }
```

2.3 View all own Invitation (After Login)



A screenshot of a REST client interface showing a **GET** method and the URL `localhost:3000/invitation/`.

2.3.1 Accept Invitation (passing params ex. /invitationId/accept) invitationId will showing in 2.3 or after sent invitation



A screenshot of a REST client interface showing a **PATCH** method and the URL `localhost:3000/invitation/6756ff1bae6aacf78c2aecc6/accept`.

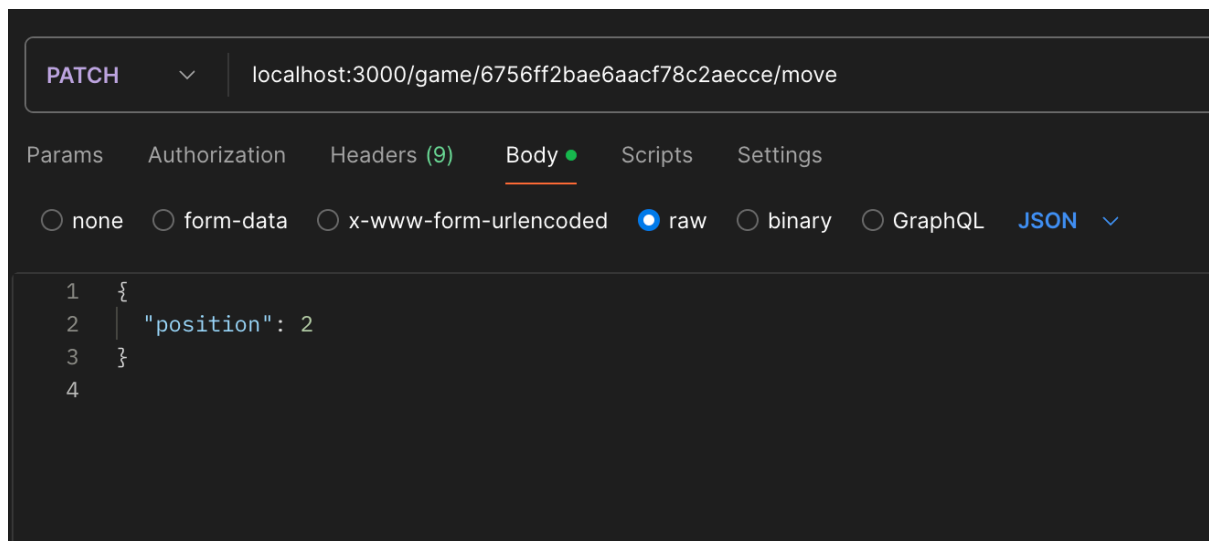
2.4 & 2.5 Get in to gameld(after accepted you will get the gameld)
and first action (if the user is the receiver in the game).
At params passing /gameid/move.
At Body passing position

Position defination

0		1		2

3		4		5

6		7		8



Guide:

Could try this by **Login1** and **Login2** Collection on **first collection**
for playing this game or function

it will show if your turn or not when patching

2.6 History & Statistic

View Stat

GET



localhost:3000/statistic

View History

GET



localhost:3000/user/history

Extra Point

User will get the gameId by using this

POST



localhost:3000/game/playwithbot/

Then user have to get in the game by passing
params: /playwithbot/gameId/move

Body: position

PATCH



localhost:3000/game/playwithbot/6756ff69ae6aacf78c2aed02/move

Params

Authorization

Headers (9)

Body

Scripts

Settings

☐ none

☐ form-data

☐ x-www-form-urlencoded

☒ raw

☐ binary

☐ GraphQL

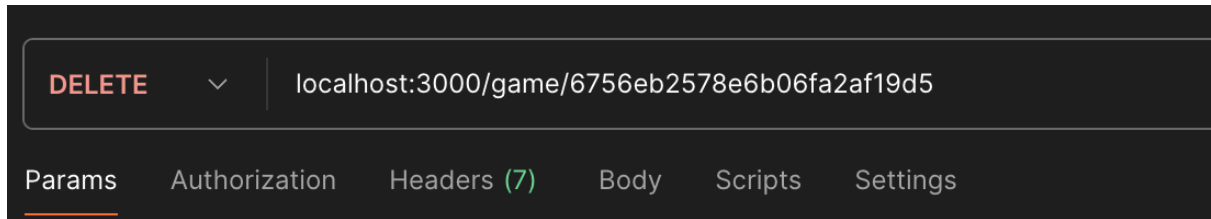
JSON



```
1 {  
2   | "position": 6  
3   }  
4
```

Extra Command

This will disconnect the game but not delete by passing gameId



Postman WorkSpace Invitation:

https://app.getpostman.com/join-team?invite_code=c366e02704fe9bb121c5327c783d1ccd&target_code=2e3e7d179a164c32e3e64c22eb570263

How to Run the project

- git clone this repository
- add .env file
- cd to the project (nest js folder) and npm i
- npm run start:dev