**Institut Sains dan Teknologi Terpadu Surabaya**

Jl. Ngagel Jaya Tengah 73 - 77, Surabaya 60284

Telp. (031) 5027920 Fax. (031) 5041509

Laboratorium : L-304

Waktu : Kamis / 10.45-12.45

Minggu Ke : 6

Materi : Authentication & Authorization

Praktikum : Web Service

Jurusan : S1 – Informatika

Tanggal : 13 April 2023

Jenis Soal : Materi dan Tugas

# MATERI (TOTAL = 40)

Wajib berikan response code: **400** jika bad request, **404** jika not found, **200** jika sukses, **201** jika created atau berhasil menambahkan data.

**Note:**

Karena materi minggu 6 merupakan Authentication dan Authorization, tiap endpoint yang membutuhkan **Authentication** berupa token sudah diberi tag **<Authentication Required>**. Apabila endpoint dengan tag ini dihit tanpa diberikan token, berikan **response error 400 dengan message “Authentication Required”. JWT Token diberikan pada request header dengan request key ‘x-auth-token’.**

Buatlah sebuah web service untuk aplikasi chatting ***sederhana***. Berikut merupakan endpoint yang perlu dibuat:

1. **[POST] /api/users**

Endpoint yang digunakan untuk meregistrasi user baru. Data yang diperlukan untuk registrasi user:

* username (case sensitive)
* name
* password

Berikan pengecekan tiap data berupa:

* username unik
* semua data diisi

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Body  - username: "mikeohearn"  - name: "Mike O’Hearn"  - password: “donthearnme” | Status: 201  Body:  {  "username": "mikeohearn",  "name ": "Mike O’Hearn"  } |
| Body  - username: "mikeohearn"  - name: "Mike O’Hearn"  - password: “donthearnme” | Status: 400  Body:  {  "message”: “Username is already taken”  } |
| Body  - username: "mikeohearn" | Status: 400  Body:  {  "message”: “Please fill all required forms”  } |

1. **[POST] /api/login**

Endpoint yang digunakan untuk login. Data yang perlu diberikan untuk melakukan login adalah username dan password. Berikan pengecekan bahwa password yang diberikan benar milik username yang diberikan. Apabila kredensial sudah benar, kembalikan JWT Token yang menyimpan informasi **username** yang sedang login dengan **secret key** “KimChaewon”.

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Body  - username: "mikeohearn"  - password: “donthearnme” | Status: 200  Body:  {  "username": "mikeohearn",  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6Im1pa2VvaGVhcm4ifQ.i1-Ko7x\_YZy2iMsoPJMqweYmnYz4qhiJZmwZlXmNFC8"  } |
| Body  - username: "mikeohearn"  - password: “donthurtme” | Status: 400  Body:  {  "message”: “Incorrect password”  } |

1. **[POST] /api/groups <Authentication Required>**

Endpoint yang digunakan untuk membuat group chat baru. Group chat menerapkan sistem role di mana user pembuat grup secara default memiliki role “Admin”. Admin dapat mengeluarkan atau menambah member grup. Untuk grup yang baru saja dibuat, generate sebuah ID grup “GR” + 3 digit nomor urut.

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Body  - name: "Ratio Squad" | Status: 201  Body:  {  “group\_id”: “GR001”,  "message": "The group `Ratio Squad` has been successfully created",  “members”: [  {  “username”: “mikeoharn”,  “name”: “Mike O’Hearn”,  “role”: “Admin”,  }  ]  } |

1. **[PUT] /api/groups/:group\_id <Authentication Required>**

Endpoint yang digunakan untuk memodifikasi grup. Modifikasi dilakukan dengan menambah atau mengeluarkan member. Untuk menambah, **type** diisi dengan “add”. Sedangkan untuk mengeluarkan, **type** diisi dengan “kick”. Data terakhir yang dibutuhkan adalah **usernames** yang berisi array of username milik akun user. Jika sudah berhasil melakukan action, kembalikan pesan bahwa action berhasil dilakukan dan list of member group saat ini.

Endpoint ini membutuhkan autorisasi yang memastikan bahwa user yang mengakses endpoint ini pada suatu grup merupakan Admin dari grup tersebut. Apabila akun yang mengakses bukan Admin dari grup yang bersangkutan: **Kembalikan error code 403** dengan message **“Unauthorized access”.** Lakukan juga pengecekan bahwa semua **usernames:**

* Teregistrasi
* Apabila dikeluarkan, pastikan berada di dalam grup
* Apabila ditambahkan ke grup, pastikan sedang tidak berada dalam grup

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Body  - type: "kick"  - usernames: [  “jovkhin”,  “pudovkin”  ] | Status: 200  Body:  {  “group\_id”: “GR001”,  “message”: “Successfully removed from group”  “members”: [  {  “username”: “mikeoharn”,  “name”: “Mike O’Hearn”,  “role”: “Admin”,  },  {  “username”: “gnomenoises”,  “name”: “Gary Barry”,  “role”: “Member”,  },  ...  ]  } |
| Body  - type: "kick"  - usernames: [  “jovkhin”,  “pudovkin”  ] | Status: 400  Body:  {  “group\_id”: “GR001”,  “message”: “Some usernames are not in the group”  } |
| Body  - type: "add"  - usernames: [  “ymd”,  “dannydoe”,  “johndoe”  ] | Status: 200  Body:  {  “group\_id”: “GR001”,  “message”: “Successfully added to group”  “members”: [  {  “username”: “mikeoharn”,  “name”: “Mike O’Hearn”,  “role”: “Admin”,  },  {  “username”: “gnomenoises”,  “name”: “Gary Barry”,  “role”: “Member”,  },  ...  ]  } |
| Body  - type: "add"  - usernames: [  “ymd”,  “dannydoe”,  “johndoe”  ] | Status: 400  Body:  {  “group\_id”: “GR001”,  “message”: “Some usernames are already in the group”  } |

1. **[POST] /api/groups/:group\_id <Authentication Required>**

Endpoint ini digunakan untuk melakukan chat dalam grup. Data yang dibutuhkan hanyalah sebuah string **message**. Jika message berhasil dikirim, kembalikan data berupa 5 chat terbaru. Tiap message terdiri atas **name, message,** dan **datetime** dengan format DD-MM-YYYY HH:mm:ss.

Jangan lupa untuk memberi autorisasi yang memastikan bahwa user yang menambahkan chat baru merupakan anggota dari group tersebut. Apabila user yang menambah chat bukan merupakan member dari grup yang bersangkutan: **Kembalikan error code 403** dengan message **“Unauthorized access”.**

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Body  - message: "what’s good my g?" | Status: 201  Body:  {  "message": "Successfully sent",  "recent\_chats”: [  {  “name”: “Mike O’Hearn”,  “message”: “what’s good my g?”,  “datetime”: 07-04-2023 14:59:21  },  {  “name”: “Miles Kilometers”,  “message”: “Afternoon y’all”,  “datetime”: 07-04-2023 14:53:31  },  ...  ]  } |

1. **[GET] /api/groups/:group\_id <Authentication Required>**

Endpoint yang digunakan untuk mengecek chat terbaru dari grup. Apabila endpoint ini dihit, tampilkan history chat dari grup tersebut mulai dari yang terbaru hingga terlama. Data yang ditampilkan berupa **message, display\_name,** dan **datetime** dengan format DD-MM-YYYY HH:mm:ss. Jangan lupa untuk memberi autorisasi yang memastikan bahwa user yang menambahkan chat baru merupakan anggota dari grup tersebut. Apabila user yang menambah chat bukan merupakan member dari grup yang bersangkutan: **Kembalikan error code 403** dengan message **“Unauthorized access”.**

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Body | Status: 200  Body:  {  "group\_name": "That One Group Chat",  "chats [  {  “name”: “Connor”,  “message”: “I’m down for some games”,  “datetime”: 07-04-2023 18:21:21  },  {  “name”: “Danny Dantes”,  “message”: “Anybody up for some ranked later?”,  “datetime”: 07-04-2023 18:12:45  },  ...  ]  } |

**Struktur Database Minggu 6 (PRAK\_M6\_WS):**

|  |  |
| --- | --- |
| users | Keterangan |
| username | varchar, PK |
| name | varchar, NN |
| password | varchar, NN |

|  |  |
| --- | --- |
| members | Keterangan |
| username | varchar, FK(users, username) |
| group\_id | varchar, FK(groups, id) |
| role | varchar, NN |

|  |  |
| --- | --- |
| groups | Keterangan |
| id | varchar, PK |
| name | varchar, NN |

|  |  |
| --- | --- |
| group\_chats | Keterangan |
| id | int, auto inc |
| username | varchar, FK(users, username) |
| name | varchar |
| group\_id | varchar, FK(groups, id) |
| message | varchar |
| datetime | datetime |

**DILARANG MENGGUNAKAN MATERI YANG BELUM DIAJARKAN PADA MINGGU INI**

**JIKA MELANGGAR MAKA NILAI MATERI : 0**

**PERHATIKAN KETENTUAN DI BAWAH:**

* **Highlight kriteria yang dikerjakan dengan warna kuning dan kumpulkan word beserta dengan file materi, apabila tidak dikumpulkan maka materi tidak akan diperiksa.**
* **Akan ada pengurangan nilai sebesar -5 untuk setiap kriteria yang dihighlight namun tidak dikerjakan.**
* **MENCONTEK = Nilai MOD 2**

## MATERI : 40

|  |  |
| --- | --- |
| **SCORE** | **KRITERIA** |
| 0/2/3 | Endpoint [POST] /api/users berjalan dengan sesuai (endpoint, field, dan respon sesuai), terdapat pengecekkan sesuai ketentuan, dan dapat mendaftarkan user baru. |
| 0/3/4 | Endpoint [POST] /api/login dapat mengembalikan JWT Token dan melakukan pengecekan kredensial dengan kembalian yang sesuai dengan soal |
| 0/2 | Endpoint [POST] /api/groups dapat membuat grup baru dengan kembalian yang sesuai dengan soal |
| 0/1/2/5/8 | Endpoint [PUT] /api/groups/:group\_id dapat memodifikasi group member dengan kembalian yang sesuai dengan soal  @1 Untuk tiap pengecekan yang dikerjakan (Max: 2) |
| 0/2/4 | Endpoint [POST] /api/groups/:group\_id dapat memposting chat baru dan mengembalikan data 5 chat terbaru dengan benar dan berformat sesuai dengan soal |
| 0/2/4 | Endpoint [GET] /api/groups/:group\_id dapat mengembalikan data semua history chat dari group dengan benar dan berformat sesuai dengan soal |
| Authentication | |
| 0/2/4/6/8 | Autentikasi untuk mengecek token berjalan dengan benar |
| Authorization | |
| 0/2 | Autorisasi pengecekan pengakses merupakan admin grup berjalan dengan benar |
| 0/2/4 | Autorisasi pengeeckan pengakses merupakan member grup berjalan dengan benar |
| MATERI: 40 | |

# TUGAS (TOTAL = 30)

Buatlah sebuah web service Discord server chat di mana tiap **server** merupakan sebuah grup yang memiliki 1 atau lebih **text channel** (chat room). Tiap server secara defualt memiliki role **Admin** yang dapat mengakses semua text channel pada suatu server. User yang membuat server secara default merupakan Admin dari server tersebut, sedangkan user yang baru saja bergabung dengan server memiliki role **Member**. Tiap server juga dapat menambah custom role. Secara default, user yang dapat menambah dan assign role pada server member merupakan **Admin.**

Tiap text channel dalam sebuah server memiliki konfigurasi yang menentukan **role** apa saja dalam server tersebut yang dapat mengakses isi text channel. Konfigurasi ini dapat diatur pada sebuah endpoint di mana hanya **Admin** yang dapat mengakses endpoint ini.

Karena tiap role memiliki IDnya masing-masing, role **Admin** memiliki ID SR000RL000. Sedangkan **Member** memiliki ID SR000RL001. Kedua role ini merupakan **reserved role** yang namanya tak dapat digunakan pada role baru yang hendak dibuat.

Berikut merupakan endpoint yang harus dibuat:

1. **[POST] /api/login**

Endpoint untuk melakukan login. Endpoint ini membutuhkan username dan password. Apabila username belum terdaftar pada saat melakukan login, otomatis daftarkan user tersebut. Apabila sudah terdaftar, pastikan password yang dimasukkan akun tersebut benar. Jika kredensial sudah sesuai, kembalikan JWT Token dengan key “SonSeungWan” yang menyimpan informasi user yang melakukan login.

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Body  - username: "mikeohearn"  - password: “donthearnme” | Status: 200  Body:  {  "username": "mikeohearn",  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6Im1pa2VvaGVhcm4ifQ.i1-Ko7x\_YZy2iMsoPJMqweYmnYz4qhiJZmwZlXmNFC8"  } |
| Body  - username: "mikeohearn"  - password: “donthurtme” | Status: 400  Body:  {  "message”: “Incorrect password”  } |

1. **[POST] /api/servers <Authentication Required>**

Endpoint yang digunakan untuk membuat server baru. Endpoint ini membutuhkan data berupa **name** yang menjadi nama server. Untuk server yang baru terbuat, generate sebuah server ID “SR+<3 digit nomor urut>”. Generate juga sebuah server code yang digunakan user untuk bergabung dengan server. Server code terdiri atas 5 karakter alphanum yang bersifat unik dan case-sensitive.

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Body  - name: "Daily Dank Memes" | Status: 201  Body:  {  server\_id: “SR001”,  name: “Daily Dank Memes”,  join\_code: “d8Lp2”  } |

1. **[POST] /api/join/:server\_code <Authentication Required>**

Endpoint yang digunakan untuk bergabung dengan server dengan menggunakan server code. Berikan pengecekan bahwa server code tersebut valid (terdaftar milik sebuah server). Sebuah server memiliki ban list. Apabila user yang ingin bergabung pada server termasuk dalam ban list, kembalikan pesan khusus yang mengindikasikan bahwa user tersebut tidak dapat bergabung dalam server.

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Param   * server\_code: “d8Lp2” | Status: 200  Body:  {  message: “You’ve successfully joined the server ‘Daily Dank Memes’”  } |
| Param   * server\_code: “qwert” | Status: 404  Body:  {  message: “The server join code doesn’t exist”  } |
| Param   * server\_code: “d8Lp2” | Status: 403  Body:  {  message: “You are currently being banned from joining this group”  } |

1. **[POST] /api/roles <Authentication Required>**

Endpoint yang digunakan untuk membuat role baru pada sebuah server. Berikut data yang dibutuhkan pada endpoint ini:

* server\_id
* role\_name

Untuk role yang baru saja terbuat, berikan role ID dengan format berikut: <Server ID>RL<3 digit nomor urut>. Berikan juga autorisasi bahwa hanya Admin dari group tersebut yang hanya dapat mengakses endpoint ini. Apabila akun yang mengakses bukan Admin dari grup yang bersangkutan: **Kembalikan error code 403** dengan message **“Unauthorized access”.**

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Body   * server\_id: “SR001” * role\_name: “Moderator” | Status: 201  Body:  {  role\_id: “SR001RL001”,  message: “The role ‘Moderator’ has been created”  } |

1. **[POST] /api/text-channels <Authentication Required>**

Endpoint digunakan untuk membuat text channel baru pada server. Data yang dibutuhkan untuk membuat text channel baru:

* server\_id (string, required)
* channel\_name (string, required)
* allowed\_roles (array, optional)

**Allowed\_roles** merupakan data yang berisi array of role ID yang dapat mengakses dan mengirim message pada channel yang hendak dibuat. Apabila tidak diisi, maka secara default channel baru dapat diakses oleh role Member. Untuk text channel yang baru terbuat, beri ID dengan format berikut: <Server ID>TC<3 digit nomor urut>. Berikan juga autorisasi bahwa hanya Admin dari group tersebut yang hanya daapt mengakses endpoint ini. Apabila akun yang mengakses bukan Admin dari grup yang bersangkutan: **Kembalikan error code 403** dengan message **“Unauthorized access”.**

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Body   * server\_id: “SR001” * channel\_name: “general” | Status: 201  Body:  {  text\_channel\_id: “SR001TC001”,  message: “Text channel `general` has been created”,  role\_access: [  {  role\_id: “SR000RL001”,  role\_name: “Member”  }  ]  } |
| Body   * server\_id: “SR001” * channel\_name: “the-council” * allowed\_roles: [   “SR001RL001”,  “SR001RL002”  ] | Status: 201  Body:  {  text\_channel\_id: “SR001TC001”,  message: “Text channel `the-council` has been created”,  role\_access: [  {  role\_id: “SR001RL001”,  role\_name: “Moderator”  },  {  role\_id: “SR001RL002”,  role\_name: “Meme Expert”  }  ]  } |

1. **[PUT] /api/text-channels/:text\_channel\_id <Authentication Required>**

Endpoint ini digunakan untuk mengedit konfigurasi role yang dapat mengakses text channel. Berikut merupakan request yang dapat diterima:

* add (array of string, optional)
* remove (array of string, optional)
* set (array of string, optional)

Ketiga key di atas menerima value berupa array of role ID yang dimiliki oleh server yang bersangkutan. Role yang berada dalam list **add** dapat mengakses text channel. Sedangkan role yang berada di dalam list **remove** tidak lagi dapat mengakses text channel. Apabila endpoint menerima key **set,** abaikan key **add** dan **remove.** Ganti semua role yang dapat mengakses text channel menjadi array of role yang menjadi nilai **set.**

Berikan pengecekan bahwa **role** yang terdapat pada ketiga request key di atas berada dalam server. Berikan juga autorisasi bahwa hanya admin yang dapat mengakses endpoint ini. Apabila akun yang mengakses bukan Admin dari grup yang bersangkutan: **Kembalikan error code 403** dengan message **“Unauthorized access”.**

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Param:   * text\_channel\_id: “SR001TC001”   Body   * add: [   “SR001RL004”,  “SR001RL007”  ] | Status: 200  Body:  {  text\_channel: “cursed-discussion”,  message: “The following role has been added to channel access: `Sigma`, `Troll`”,  role\_access: [  {  role\_id: “SR000RL001”,  role\_name: “Moderator”  },  {  role\_id: “SR000RL003”,  role\_name: “Dark Lord”  },  {  role\_id: “SR000RL004”,  role\_name: “Sigma”  },  ...  ]  } |
| Param:   * text\_channel\_id: “SR001TC001”   Body   * remove: [   “SR000RL001”  ] | Status: 200  Body:  {  text\_channel: “cursed-discussion”,  message: “The following role has been removed from channel access: `Member`”,  role\_access: [  {  role\_id: “SR000RL001”,  role\_name: “Moderator”  },  {  role\_id: “SR000RL003”,  role\_name: “Dark Lord”  },  {  role\_id: “SR000RL004”,  role\_name: “Sigma”  },  ...  ]  } |
| Param:   * text\_channel\_id: “SR001TC001”   Body   * set: [   “SR001RL001”  ] | Status: 200  Body:  {  text\_channel: “ban-discussion”,  message: “Role access for this channel has been completely replaced”  role\_access: [  {  role\_id: “SR000RL001”,  role\_name: “Moderator”  }  ]  } |

1. **[PUT] /api/servers/:server\_id/assign <Authentication Required>**

Endpoint ini digunakan untuk mengassign role pada sebuah member dari server. Data yang dibutuhkan untuk endpoint ini:

* username
* role\_id

Penambahan role bersifat toggle, apabila user yang diberi role X sudah memiliki role tersebut, hilangkan role dari user. Sebaliknya apabila user belum memiliki role tersebut, tambahkan role pada user.

Berikan pengecekan bahwa **role\_id** terdapat pada server dan username berada dalam server. Berikan juga autorisasi bahwa user yang dapat mengakses endpoint ini merupakan **Admin** dari server. Apabila akun yang mengakses bukan Admin dari grup yang bersangkutan: **Kembalikan error code 403** dengan message **“Unauthorized access”.**

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Param:   * server\_id: “SR001”   Body   * username: “dragonslz” * role\_id: “SR001RL007” | Status: 200  Body:  {  message: “The role `Troll` has been added to user”  } |
| Param:   * server\_id: “SR001”   Body   * username: “dragonslz” * role\_id: “SR001RL007” | Status: 200  Body:  {  message: “The role `Troll` has been removed from user”  } |
| Param:   * server\_id: “SR001”   Body   * username: “dragonslayer” * role\_id: “SR001RL007” | Status: 404  Body:  {  message: “User isn’t a part of server member”  } |
| Param:   * server\_id: “SR001”   Body   * username: “dragonslz” * role\_id: “SR001RL420” | Status: 404  Body:  {  message: “The server doesn’t have the given role”  } |

1. **[PUT] /api/servers/:server\_id/members <Authentication Required>**

Endpoint yang digunakan untuk memodifikasi daftar user dalam server. Terdapat dua action yang dapat dilakukan, yaitu **kick** dan **ban.** User yang terkena action **kick** akan langsung dikeluarkan dari server. Sedangkan user yang terkena action **ban** akan langsung dikeluarkan dari server dan tidak dapat bergabung dalam server tersebut lagi. Data yang diperlukan untuk endpoint ini:

* action (string, required, [kick, ban])
* usernames (array of string, required)

Berikan pengecekan bahwa username dalam **usernames** berada dalam server. Berikan juga autorisasi bahwa hanya **Admin** dari grup yang dapat mengakses endpoint ini. Apabila akun yang mengakses bukan Admin dari grup yang bersangkutan: **Kembalikan error code 403** dengan message **“Unauthorized access”.**

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Param:   * server\_id: “SR001”   Body   * action: “kick” * usernames: [   “CohnJena”,  “TheExiled”  ] | Status: 200  Body:  {  message: “Users have been successfully kicked”  } |
| Param:   * server\_id: “SR001”   Body   * action: “ban” * usernames: [   “ImGonnaSaytheNWord”,  “ElRacista”  ] | Status: 200  Body:  {  message: “Users have been successfully banned”  } |
| Param:   * server\_id: “SR001”   Body   * action: “ban” * usernames: [   “gooooose”,  “Stonks”  ] | Status: 404  Body:  {  message: “Some users aren’t a part of server member”  } |

1. **[POST] /api/text-channels/:text\_channel\_id <Authentication Required>**

Endpoint yang digunakan untuk mengirimkan chat pada sebuah text channel. Jika berhasil mengirimkan pesan, kembalikan history chat dari text channel tersebut dari yang terbaru hingga terlama. Sebelum mengirimkan pesan, berikan pengecekan autorisasi bahwa hanya user yang memiliki role yang sudah ditentukan dapat mengirimkan chat pada text channel.

|  |  |
| --- | --- |
| **Contoh Request** | **Contoh Response** |
| Param:   * text\_channel\_id: “SR001TC001”   Body   * text: “My honest reaction: :)” | Status: 200  Body:  {  chats: [  {  username: “mikeohearn”,  message: “My honest reaction: :)”,  datetime: 08-04-2023 23:25:42  },  ...  ]  } |
| Param:   * text\_channel\_id: “SR001TC001”   Body   * text: “My honest reaction: :)” | Status:403  Body:  {  message: “Lu itu ga diajak :(”  } |

**Sediakan database dengan ketentuan berikut:**

1. **Memiliki 1 server**
2. **Server tsb memiliki minimal 3 user**
3. **User A sbg Admin, User B sbg <Custom Role>, dan User C sbg Member**
4. **Server tsb minimal memiliki 1 text channel**

**WAJIB MENGGUNAKAN DATABASE DAN SEQUELIZE**

**DILARANG MENGGUNAKAN KONSEP MAUPUN MATERI YANG BELUM DIAJARKAN**

**JIKA MELANGGAR MAKA NILAI TUGAS: 0**

**PERHATIKAN KETENTUAN DI BAWAH:**

* **Highlight kriteria yang dikerjakan dengan warna kuning dan kumpulkan word beserta dengan file tugas, apabila tidak dikumpulkan maka tugas tidak akan diperiksa.**
* **Akan ada pengurangan nilai sebesar -5 untuk setiap kriteria yang dihighlight namun tidak dikerjakan.**
* **Wajib menghapus FOLDER “NODE\_MODULES” sebelum mengumpulkan! Melanggar = Nilai Tugas = 0**
* **Nama endpoint HARUS SAMA dengan ketentuan SOAL. Endpoint yang berbeda menybebkan endpoint tersebut tidak akan diperiksa.**
* **Nama Field dan Response harus sama dengan ketentuan setiap soal. Melanggar = nilai kriteria tersebut div 2**
* **Wajib menggunakan port 3000**
* **JANGAN LUPA MENGUMPULKAN DATABASE**
* **Nama Database : T6\_4DigitNRP**

## TUGAS : 30

|  |  |
| --- | --- |
| **SCORE** | **KRITERIA** |
| 0/1/2 | Endpoint [POST] /api/login berjalan dengan sesuai (endpoint, field, dan respon sesuai), soal, meregistrasikan user baru ketika pertama kalo login, dan mengembalikan JWT Token |
| 0/1/2 | Endpoint [POST] /api/servers dapat membuat server baru dengan respon yang sesuai dengan soal |
| 0/1/2 | Endpoint [POST] /api/join dapat membuat user tergabung dengan sebuah server. Kembalian endpoint dan pengecekannya sesuai dengan soal |
| 0/1 | Endpoint [POST] /api/roles dapat menambah role baru dengan ketentuan dan kembalian yang sesuai dengan soal |
| 0/1/3 | Endpoint [POST] /api/text-channels dapat menambah text channel baru dengan ketentuan default role atau assigned role yang sudah diset |
| 0/2/4 | Endpoint [PUT] /api/text-channels/:text\_channel\_id dapat mengedit role yang dapat mengakses sebuah text channel. Endpoint memiliki kembalian dan pengecekan yang sesuai dengan soal |
| 0/1/2 | Endpoint [PUT] /api/servers/:server\_id/assign dapat mengganti dengan menambah atau menghapus role suatu member. Kembalian dan pengecekan endpoint ini sesuai dengan soal |
| 0/1/2/3 | Endpoint [PUT] /api/servers/:server\_id/members dapat melakukan kick atau ban server member dengan kembalian dan pengecekan yang sesuai dengan soal |
| 0/1/3 | Endpoint [POST] /api/text-channels/:text\_channel\_id dapat mengirim message dalam sebuah text channel dengan data kembalian yang benar dan sesuai dengan soal. |
| **Authentication** | |
| 0/1/2 | Mekanisme autentikasi yang membutuhkan JWT Token untuk mengakses endpoint berjalan dengan benar |
| **Authorization** | |
| 0/1/2 | Autorisasi bahwa hanya **Admin** dari server yang dapat mengakses suatu endpoint berjalan dengan benar |
| 0/2/4 | Sistem autorisasi hanya user dengan role tertentu yang dapat mengakses text channel berjalan dengan benar |
| TUGAS: 30 | |

Penyusun Soal

(Tim Asisten)

Asisten

Menyetujui

(Evan Kusuma Susanto, S.Kom., M.Kom.)

Koordinator Kuliah

Mengetahui

(Grace Levina Dewi, M.Kom.)

Koordinator Laboratorium