

**LAPORAN PROJECT UTS**  
**SISTEM MANAJEMEN IP POLSTAT STIS**



**Dosen Pengampu:**

Ibnu Santoso, S.S.T., M.T

**Disusun oleh:**

Aliefta Zulvansyah Bahyperdana      222111873 | 3SI1

**PROGRAM STUDI DIPLOMA IV KOMPUTASI STATISTIK**  
**POLITEKNIK STATISTIKA STIS**

**2023**

**BAB I**  
**PENDAHULUAN**

**1.1 Latar belakang**

Politeknik Statistika STIS (Sekolah Tinggi Ilmu Statistik) adalah sebuah lembaga pendidikan tinggi di Indonesia yang fokus pada pengembangan ilmu statistik. Di dalam konteks akademik, informasi mengenai Indeks Prestasi (IP) mahasiswa adalah salah satu aspek yang sangat penting. IP merupakan gambaran prestasi akademik seorang mahasiswa, yang mencerminkan kemampuan dan pencapaian mereka dalam mengejar gelar sarjana. Oleh karena itu, sistem manajemen IP mahasiswa di kampus Politeknik Statistika STIS adalah suatu langkah yang sangat penting dalam memenuhi kebutuhan administratif dan pengambilan keputusan yang berkaitan dengan akademik di kampus ini.

Sistem manajemen IP ini dirancang untuk 3 role yakni mahasiswa, dosen, dan admin. Mahasiswa dalam sistem ini hanya diperkenankan untuk melihat nilai, ip per semester, profil, dan mata kuliah. Dosen diberi kewenangan yang lebih banyak seperti memberi nilai kepada mahasiswa sesuai dengan matkul yang diampu. Untuk admin dalam sistem ini diberi keleluasaan dalam mengelola semua data yang ada misalnya menambahkan mataKuliah, periode, menggenerate ip per semester, crud user, mengset mata kuliah yang diampu dosen, mengset mahasiswa dalam kelas , dan masih banyak lagi.

Selain itu sistem ini mengambil refrensi perhitungan IP dari perhitungan IP yang dipakai oleh Polstat STIS. Rumus perhitungan IP yang ada di STIS adalah sebagai berikut

$$\frac{\sum(bobot\ nilai * sks\ matkul)}{\sum sks\ matkul}$$

Bobot nilai dapat dilihat dari tabel berikut

**Tabel 1 pembobotan nilai**

| No  | Rentang Nilai            | Nilai Huruf | Bobot Nilai |
|-----|--------------------------|-------------|-------------|
| (1) | (2)                      | (3)         | (4)         |
| 1   | 85,0 – 100               | A           | 4,00        |
| 2   | 80,0 – 84,9              | A-          | 3,75        |
| 3   | 75,0 – 79.9              | B+          | 3,50        |
| 4   | 70,0 – 74,9              | B           | 3,00        |
| 5   | 65,0 – 69,9              | C+          | 2,50        |
| 6   | 60,0 – 64,9              | C           | 2,00        |
| 7   | 55,0 – 59,9              | D+          | 1,50        |
| 8   | <55,0                    | D           | 1,00        |
| 9   | Komponen tidak lengkap*) | E           | 0           |

Untuk mendapatkan bobot nilai terlebih dahulu menghitung nilai angka dari tiap mata kuliah. Untuk mata kuliah kategori teori nilai angka didapatkan dengan rumus berikut:

$$0.35 * Nilai\ UTS + 0.35 * Nilai\ UAS + 0.3 * Nilai\ tugas$$

Sementara untuk mata kuliah kategori teori+praktikum, nilai angka didapatkan dengan rumus berikut:

$$0.3 * \text{Nilai UTS} + 0.3 * \text{Nilai UAS} + 0.3 * \text{Nilai Praktikum} + 0.1 \text{Nilai Tugas}$$

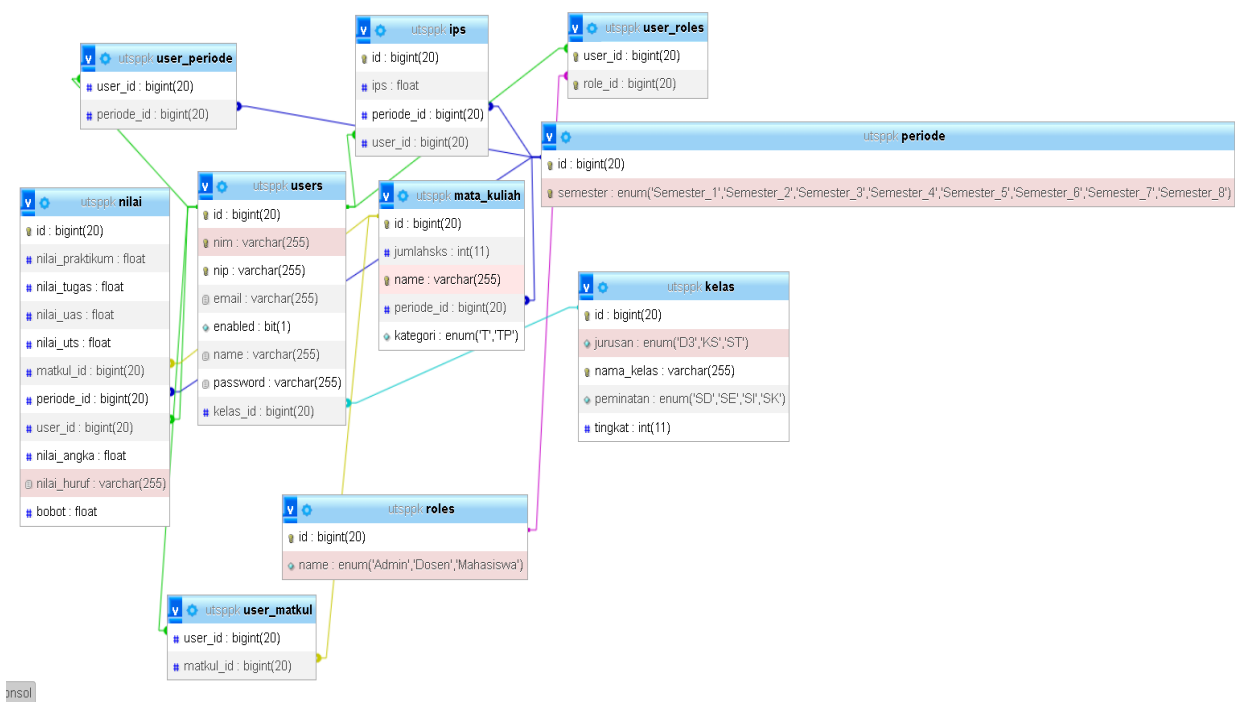
Alur proses bisnis utama dalam sistem ini adalah sebagai berikut.

- Admin membuat periode semester
- Admin membuat beberapa mata kuliah dan mengatur pada semester berapa matkul tersebut diajarkan
- Admin menetapkan dosen pengampu setiap matkul
- Mahasiswa melakukan register dan login
- Mahasiswa dapat melihat profil, mengubah profil, dan mengubah password mereka
- Mahasiswa juga dapat menghapus akun miliknya
- Admin berhak menghapus akun user
- Pada periode pengiputan nilai dosen berhak menginputkan nilai ke mahasiswa sesuai matakuliah yang diampunya. Nilai yang diinput disini adalah nilai uas, uts, tugas, dan praktikum bila kategori matkulnya teori+praktikum. Untuk nilai angka, nilai huruf serta bobot akan degenerate otomatis dari sistem.
- Dosen berhak menghapus dan juga edit nilai
- Dosen dan admin dapat melihat nilai semua mahasiswa
- Mahasiswa dapat melihat daftar nilai miliknya
- Daftar nilai disini berisi nilai uas, uts, tugas, praktikum, nilai angka, huruf, serta bobot
- Jika periode penginputan nilai sudah selesai maka admin akan melakukan generate IP per semester untuk seluruh mahasiswa
- Mahasiswa dapat melihat IP miliknya sendiri
- Admin dan dosen dapat melihat ip milik semua mahasiswa

BAB II

ISI

2.1 Entity Relationship Diagram

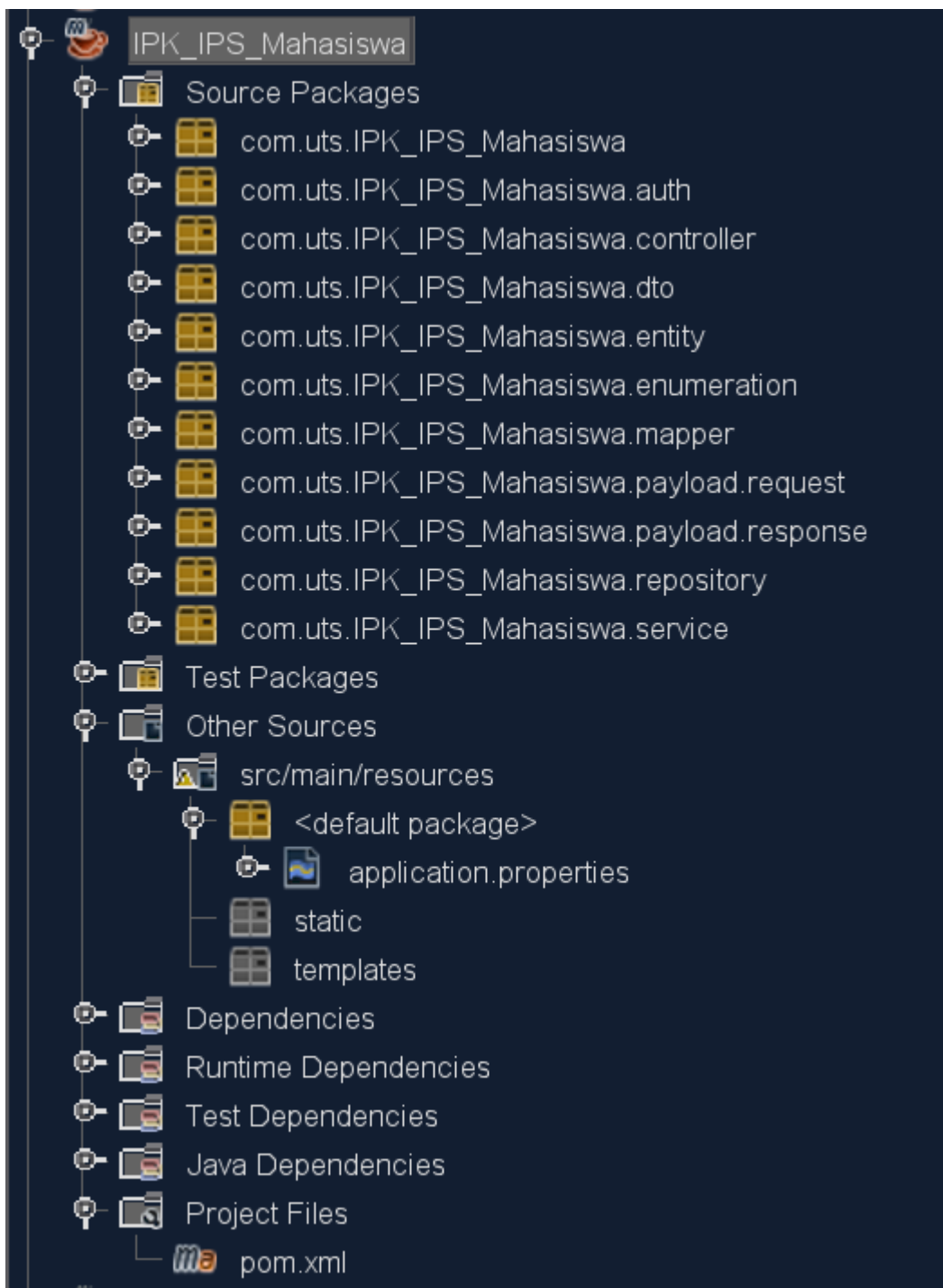


Pada ERD diatas terdapat 10 tabel.10 tabel tersebut dibuild dari 7 entity yang adadi kode program. Jadi dapat disimpulkan ada 7 tabel utama dan 3 lainnya merupakan tabel relasi manyTomany dari entity yang ada. Tabel utama yang terbentuk adalah sebagai berikut:

- User : merupakan tempat untuk menyimpan user atau pengguna dari sistem. Tabel user ini memiliki hubungan ManyToMany dengan tabel roles, periode, dan matkul. Selain itu tabel user juga memiliki hubungan OneToMany ke tabel IPS dan nilai, serta ManyToOne ke tabel kelas.
- Roles: tabel ini berfungsi menyimpan role yang dimiliki oleh user. Role yang disimpan ini hanya dibatasi 3 role yakni Mahasiswa, Dosen, dan Admin. Tiap-tiap role memiliki kewenangan tersendiri dalam sistem.
- Mata kuliah: tabel ini berfungsi menyimpan mata kuliah yang ada di Polstat STIS. Tabel ini memiliki relasi ManyToMany dengan user, relasi OneToOne dengan nilai, dan relasi ManyToOne dengan periode.
- Nilai: tabel ini berfungsi menyimpan atribut nilai serta bobot nilai dari mata kuliah yang berfungsi dalam penghitungan IP. Tabel ini memiliki hubungan ManyToOne dengan user, mata kuliah, dan periode.
- Periode: tabel ini sejatinya hanya menyimpan nama semester. Tabel ini memiliki hubungan OneToMany dengan mata kuliah , nilai, ips, serta hubungan ManyToMany dengan user.
- Kelas: tabel ini menyimpan informasi dari kelas mulai dari tingkat, jurusan, serta peminatan. Tabel ini memiliki hubungan OneToMany dengan user.
- IPS: tabel ini berfungsi menyimpan IP mahasiswa per semesternya. Tabel ini memiliki hubungan ManyToOne dengan user dan periode.

2.3 Penjelasan singkat kode program

Berikut adalah struktur package dari kode program sistem ini



Package auth berisi berbagai hal yang berkaitan dengan autentikasi meliputi JWT serta pengaturan web security. Package controller berisi sekumpulan restController yang mengatur path secara manual untuk melakukan request api. Package dto hanya berisi UserDto yang berguna untuk pertukaran data user. Package entity berisi struktur tabel yang akan dibuat dalam sistem. Package enumeration berisi sekumpulan enumeration untuk mengatur input value dari suatu kolom tabel. Package mapper hanya berisi UserMapper yang berfungsi untuk mengubah, menyambungkan, atau mentranslasikan dari entity user ke DTO atau sebaliknya. Package payload berisi sekumpulan struktur request dan response yang diperbolehkan dalam melakukan request API. Package repository disini rata rata sudah merupakan restRepository yang sudah mengset path untuk melakan request. Package repository ini umumnya digunakan untuk melakukan save ke database atau menemukan entity berdasarkan atribut variabelInnya. Package service berisi layanan layanan yang dibutuhkan dalam sistem ini.

## 2.4 Dokumentasi

Dokumentasi untuk API ini dibuat menggunakan postman dengan link file ada di sini:

<https://documenter.getpostman.com/view/27677435/2s9YR58FrC>

Untuk kode lengkap ada di github berikut:

2.5 Pengujian

Sebelum melakukan pengujian aplikasi diharapkan untuk insert table roles terlebih dahulu di database degngan query berikut

```
INSERT INTO roles(name) VALUES('Mahasiswa');
INSERT INTO roles(name) VALUES('Dosen');
INSERT INTO roles(name) VALUES('Admin');
```

2.5.1 Register

- Register Admin

UTS PPK (IPS IPK Mahasiswa) / Register / Register Admin

POST http://localhost:8080/register

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
{
  "nip": "123321123",
  "email": "Ademin@stis.ac.id",
  "name": "Ademin STeIes",
  "password": "123456",
  "role": ["Admin"]
}
```

Body Cookies Headers (14) Test Results

200 OK 1084 ms 466 B Save as example

Pretty Raw Preview Visualize JSON

```
{
  "message": "User registered successfully!"
}
```

users

Limit to 1000 rows

SELECT \* FROM utspk.users;

Result Grid

|   | id   | nim       | nip             | email                | enabled | name                 | password                  |
|---|------|-----------|-----------------|----------------------|---------|----------------------|---------------------------|
| 1 | 1    | NULL      | 132314344214124 | AgusRiyan@stis.ac.id | 0       | Agus Riyan           | \$2a\$10\$N6K90yOGB0.8sl  |
| 2 | 2    | 222111873 | NULL            | 222111873@stis.ac.id | 0       | Aliefta Zulvansyah B | \$2a\$10\$Y8M1jjYr.RWF/O: |
| 3 | 3    | NULL      | 112334242       | Surya@stis.ac.id     | 0       | Admin ganteng        | \$2a\$10\$b0AdCbH839o7Q   |
| 4 | 4    | NULL      | 123321123       | Ademin@stis.ac.id    | 0       | Ademin STeIes        | \$2a\$10\$Udh8HsGOxy07xl  |
| * | NULL | NULL      | NULL            | NULL                 | NULL    | NULL                 | NULL                      |

- Register Dosen

HTTP

UTS PPK (IPS IPK Mahasiswa) / Register / Register Dosen

Save

POST

http://localhost:8080/register

Send

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Cook

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

Beauti

1

{

2

...."nip": "2331321321323",

3

...."email": "Irfan@stis.ac.id",

4

...."name": "Irfan Jaya",

5

...."password": "123456",

6

...."role": ["Dosen"]

7

}

Body

Cookies

Headers (14)

Test Results

200 OK

1916 ms

466 B

Save as example

Pretty

Raw

Preview

Visualize

JSON

1

{

2

"message": "User registered successfully!"

3

}

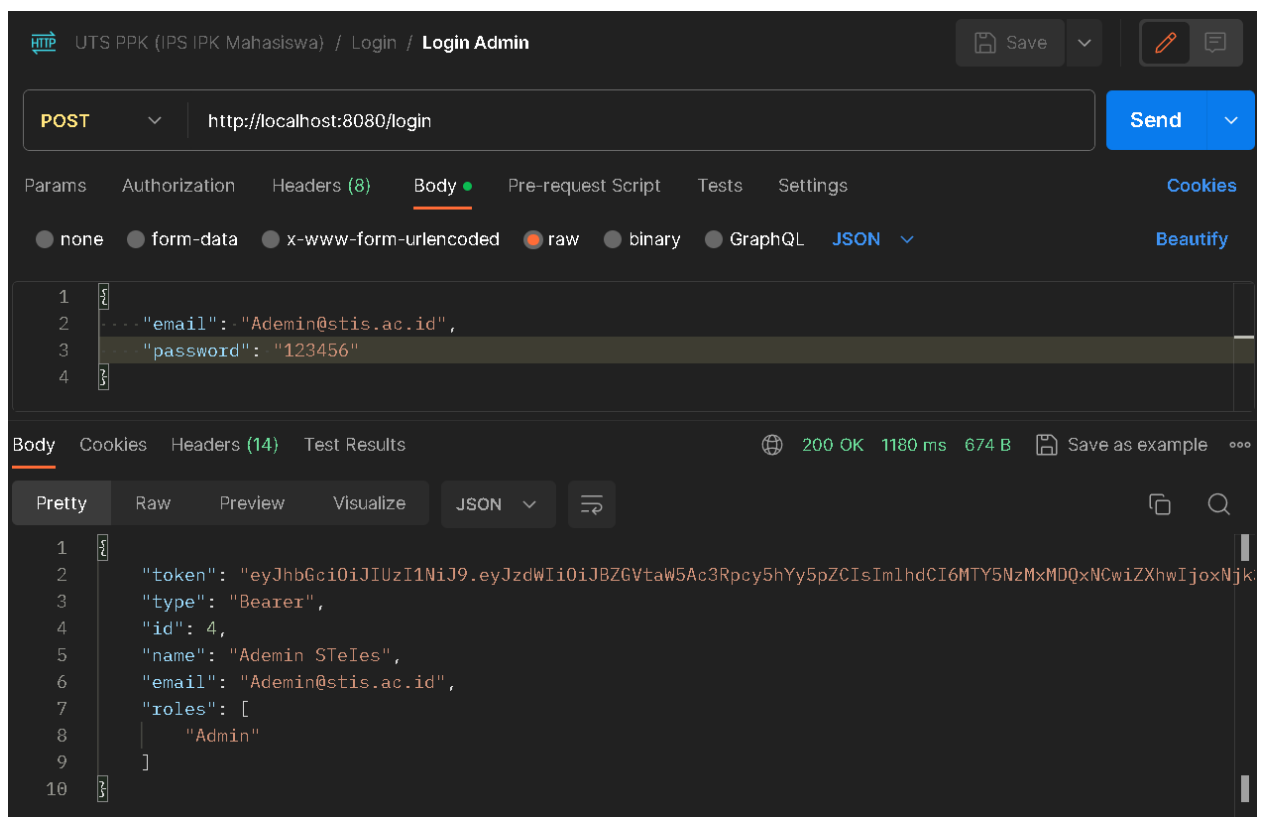
1 • `SELECT * FROM utspk.users;`

|   |      |           |                 |                      |         |                      |                           |
|---|------|-----------|-----------------|----------------------|---------|----------------------|---------------------------|
|   | id   | nim       | nip             | email                | enabled | name                 | password                  |
| ▶ | 1    | NULL      | 132314344214124 | AgusRiyan@stis.ac.id | 0       | Agus Riyan           | \$2a\$10\$N6K90yOGB0.8sl  |
|   | 2    | 222111873 | NULL            | 222111873@stis.ac.id | 0       | Aliefta Zulvansyah B | \$2a\$10\$Y8M1jjYr.RWF/O: |
|   | 3    | NULL      | 112334242       | Surya@stis.ac.id     | 0       | Admin ganteng        | \$2a\$10\$b0AdCbH839o7Q   |
|   | 4    | NULL      | 123321123       | Ademin@stis.ac.id    | 0       | Ademin STeIes        | \$2a\$10\$UdhBHsGOxy07xl  |
|   | 5    | NULL      | 2331321321323   | Irfan@stis.ac.id     | 0       | Irfan Jaya           | \$2a\$10\$Z5S99LCyRqZRRc  |
| * | NULL | NULL      | NULL            | NULL                 | NULL    | NULL                 | NULL                      |

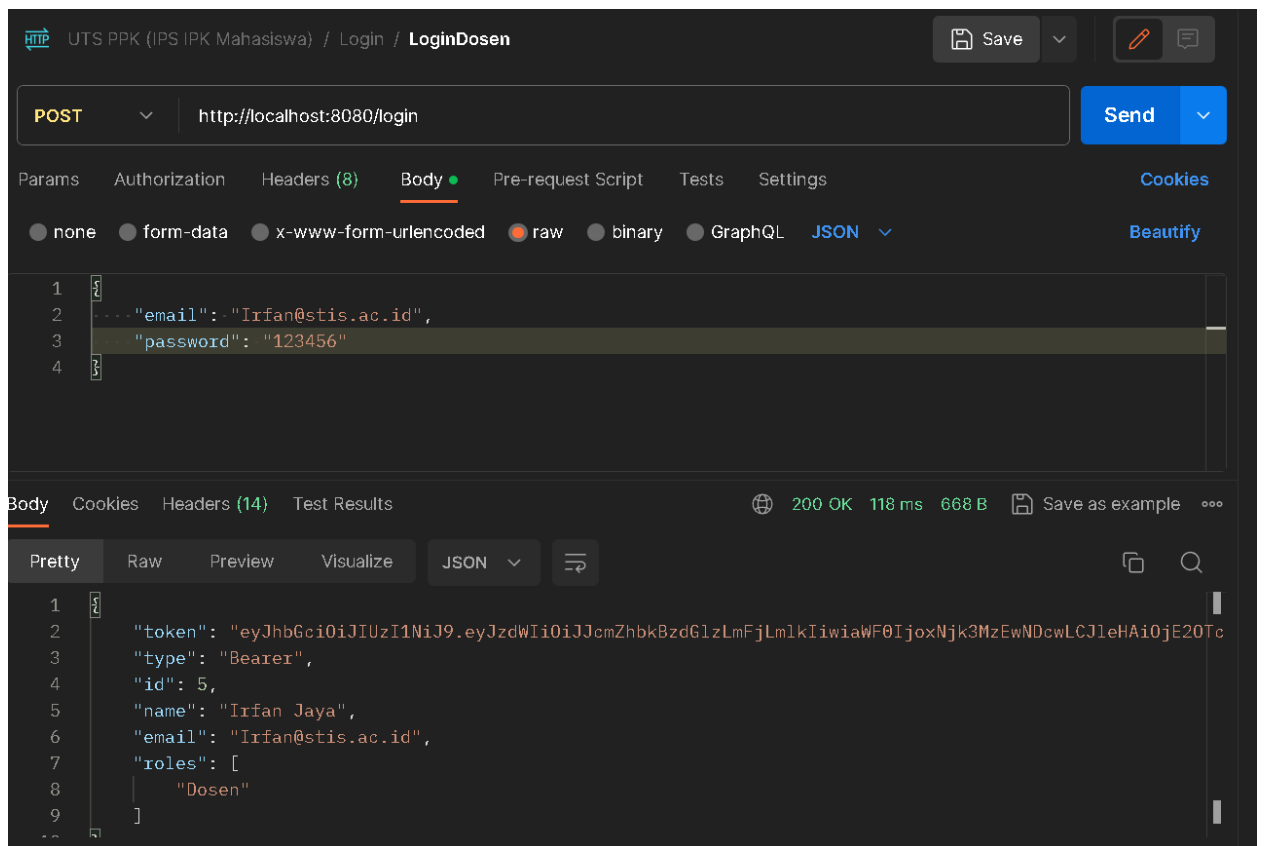
- *Regisrter Mahasiswa*



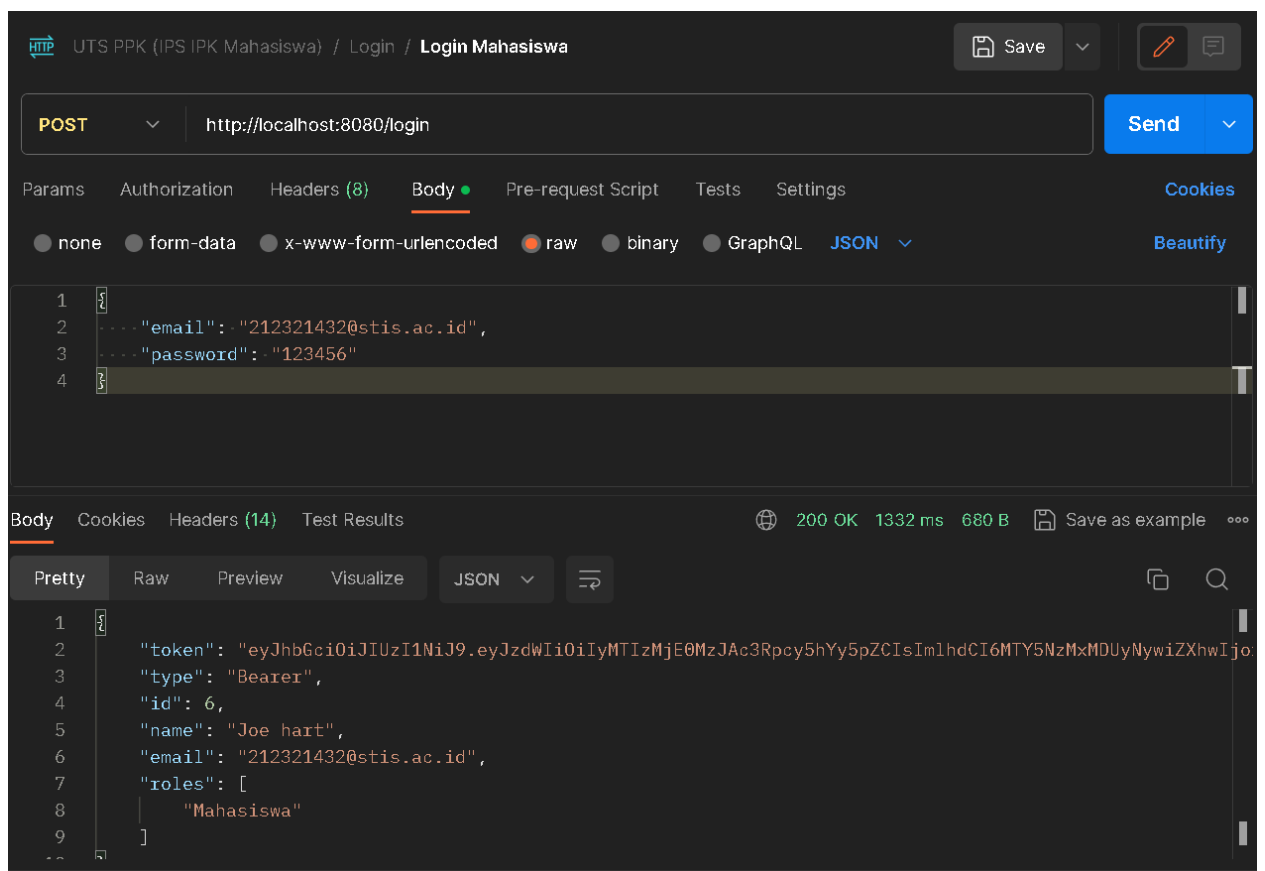




- *Login Dosen*

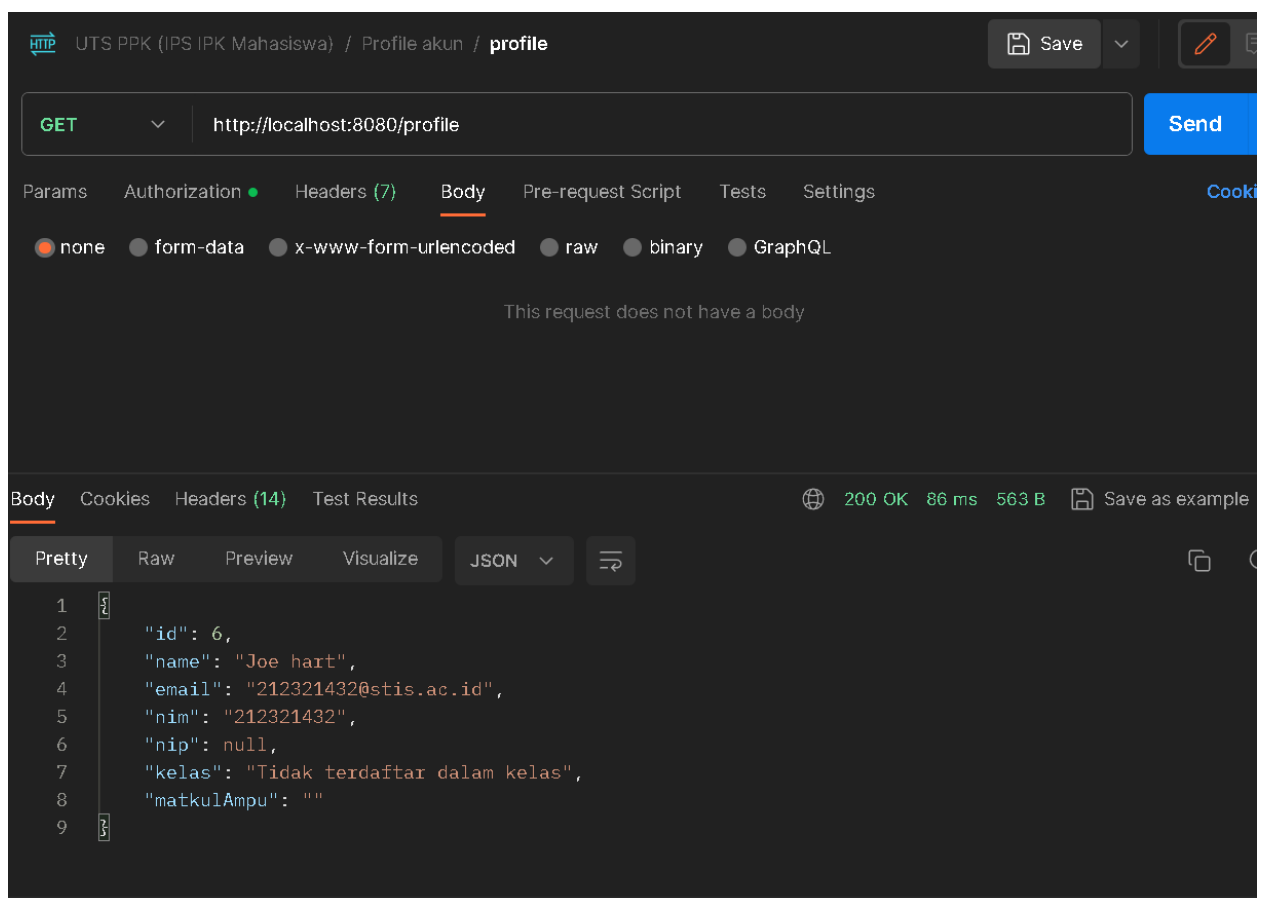


- *Login Mahasiswa*



### 2.5.3 Profile Akun

- *Lihat profile*



Path ini mengambil user dari jwtToken. Jadi otomatis dapat diketahui user yang login dan responsenya mengembalikan profile dari user yang sedang login.

- *Ubah Password*

PATCH

EditPassword

http://localhost:8080/user/changePassword

mengedit password user yang sedang login

AUTHORIZATION

Bearer Token

Token

<token>

Body

raw (json)

json

```
{
  "email": "Irfan@stis.ac.id",
  "oldPassword": "123456",
  "newPassword": "654321"
}
```

Example Request

EditPassword

curl

```
curl --location --request PATCH 'http://localhost:8080/user/ch
--data-raw '{
  "email": "Ademin@stis.ac.id",
  "oldPassword": "123456",
  "newPassword": "654321"
}'
```

View More

Example Response

Body

Headers (14)

200 OK

json

```
{
  "message": "ubah password berhasil"
}
```

Logic dari request ini adalah mengecek email yang dimasukan apakah sama dengan email user yang sedang login dengan memanfaatkan jwt token. Selanjutnya akan dicek apakah password lama yang dimasukkan sama dengan password sebelumnya. Jika iya maka ganti password berhasil dan akan muncul pesan seperti diatas.

Jika email tidak sesuai dengan user yang sedang login akan muncul pesan dibawah ini

PATCH

EditPassword

http://localhost:8080/user/changePassword

mengedit password user yang sedang login

AUTHORIZATION

Bearer Token

Token

<token>

Body

raw (json)

json

```
{
  "email": "Irfan@stis.ac.id",
  "oldPassword": "123456",
  "newPassword": "654321"
}
```

Example Request

EditPassword

curl

```
curl --location --request PATCH 'http://localhost:8080/changeP
--data-raw '{
  "email": "Ademin@stis.ac.id",
  "oldPassword": "123456",
  "newPassword": "654321"
}'
```

View More

Example Response

Body

Headers (14)

200 OK

json

```
{
  "message": "email dan user yang sedang login tidak cocok"
}
```

Sedangkan jika oldpassword yang dimasukkan tidak sesuai dengan password yang ada di database akan muncul pesan berikut

PATCH

EditPassword

http://localhost:8080/user/changePassword

mengedit password user yang sedang login

AUTHORIZATION Bearer Token

Token<token>

Body raw (json)

json

```
{
  "email": "Irfan@stis.ac.id",
  "oldPassword": "123456",
  "newPassword": "654321"
}
```

Example Request

EditPassword

curl

```
curl --location --request PATCH 'http://localhost:8080/changeP'
--data-raw '{
  "email": "222111873@stis.ac.id",
  "oldPassword": "1234s3242456",
  "newPassword": "654321"
}'
```

View More

Example Response

Body

Headers (14)

200 OK

json

```
{
  "message": "ubah password gagal"
}
```

- Edit profile

Pada request ini akan mengambil usedr yang sedang login dari jwt token lalu akan mengubah nama dari user yang sedang login tadi dengan input request name yang ada. Pada edit profile ini hanya diperkenankan mengedit nama karena untukvariabel lain dianggap tidak boleh diedit sesuka hati. Pada kasus ini akan dicoba mengedit akun mahasiswa bernama joe hart menjadi jojo. Jadi login dahulu menggunakan akun Joe hart.

Sebelum

|  |      |           |      |                      |      |          |                          |
|--|------|-----------|------|----------------------|------|----------|--------------------------|
|  | 6    | 212321432 | NULL | 212321432@stis.ac.id | 0    | Joe hart | \$2a\$10\$CGHjOQPxfP2Ghv |
|  | NULL | NULL      | NULL | NULL                 | NULL | NULL     | NULL                     |

Sesudah

HTTP

UTS PPK (IPS IPK Mahasiswa) / Profile akun / EditProfile

Save

PATCH

http://localhost:8080/profile

Send

Params

Authorization

Headers (9)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

Beautify

```
1 {
2   ... "name": "Jojo"
3 }
```

Body

Cookies

Headers (14)

Test Results

200 OK

1233 ms

462 B

Save as example

Pretty

Raw

Preview

Visualize

JSON

```
1 {
2   "message": "ubah nama profil berhasil"
3 }
```

|  |   |           |               |                      |   |            |                          |
|--|---|-----------|---------------|----------------------|---|------------|--------------------------|
|  | 5 | NULL      | 2331321321323 | Irfan@stis.ac.id     | 0 | Irfan Jaya | \$2a\$10\$Z55Y9LCyRqZRRc |
|  | 6 | 212321432 | NULL          | 212321432@stis.ac.id | 0 | Jojo       | \$2a\$10\$CGHjOQPxfP2Ghv |

- Delete Akun

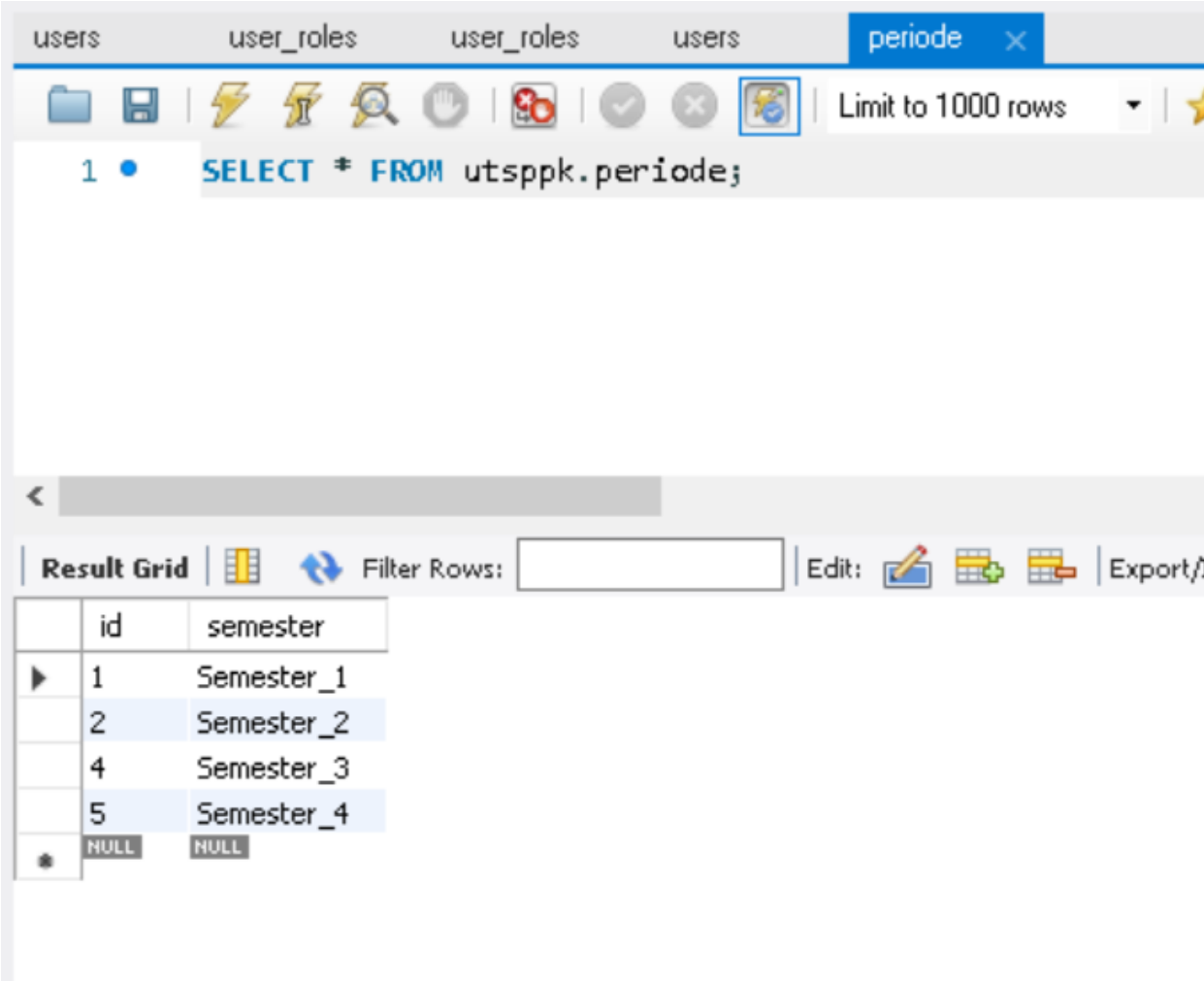
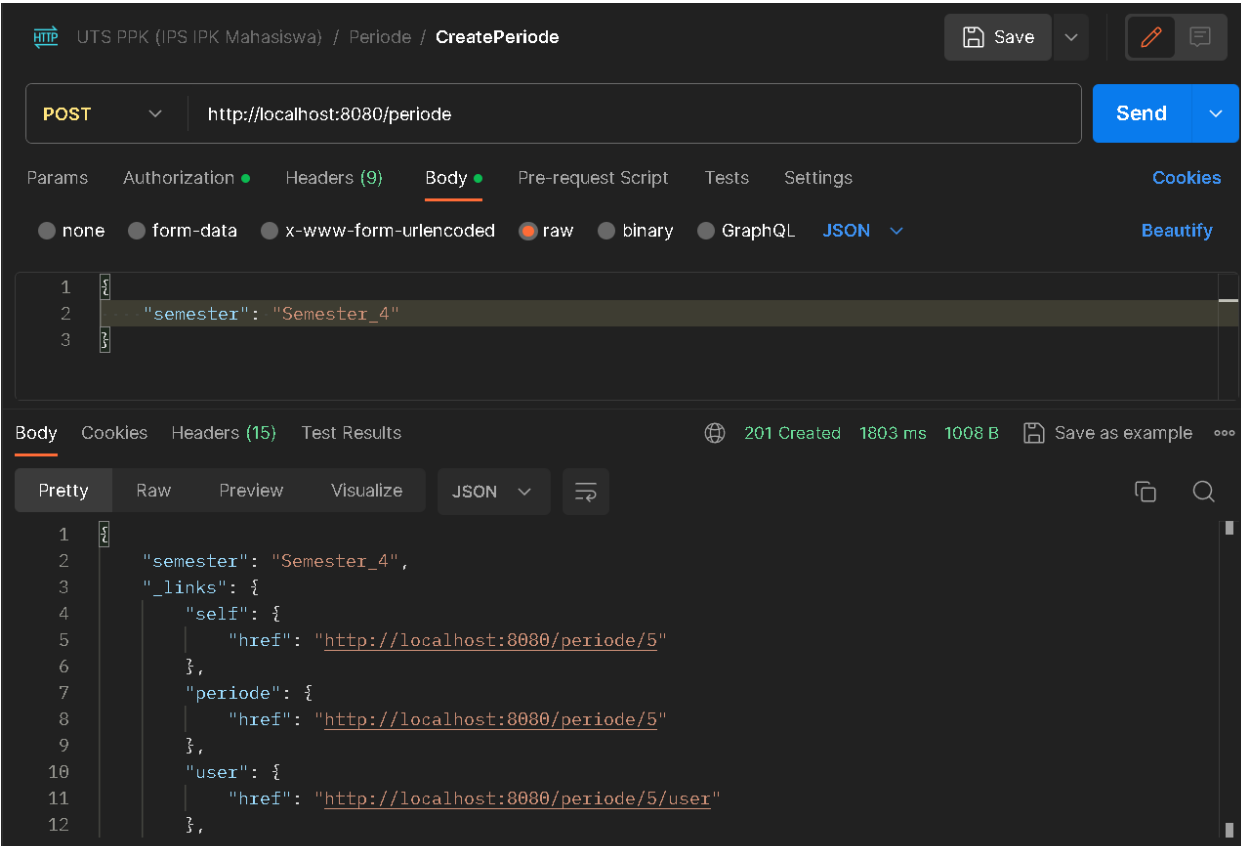
Logicnya sama dengan path profile lainnya. Akan diambil user yang sedang login dari jwt token lalu akan menghapus akun tersebut. Pada kasus ini kita akan login sebagai akun Jojo

Sebelum

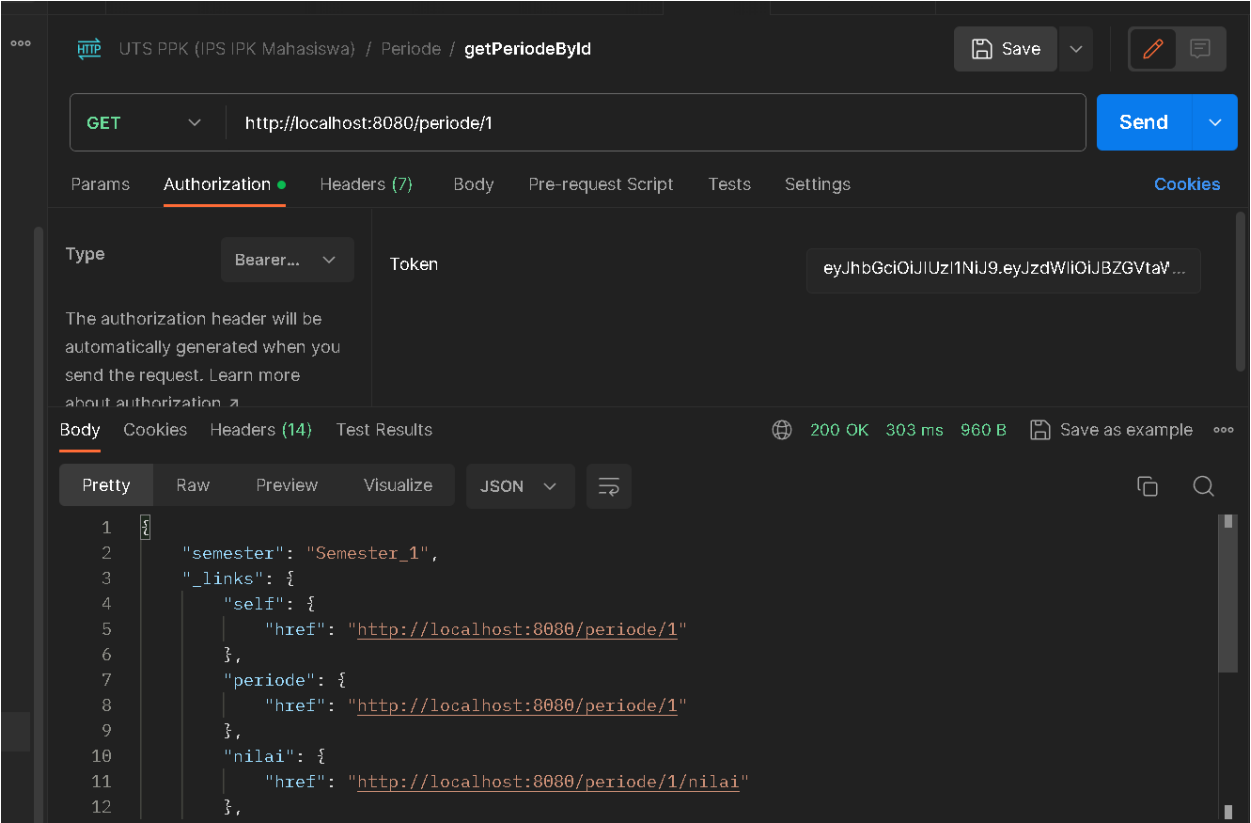
|  |   |           |               |                      |   |            |                          |
|--|---|-----------|---------------|----------------------|---|------------|--------------------------|
|  | 5 | NULL      | 2331321321323 | Irfan@stis.ac.id     | 0 | Irfan Jaya | \$2a\$10\$Z55Y9LCyRqZRRc |
|  | 6 | 212321432 | NULL          | 212321432@stis.ac.id | 0 | Jojo       | \$2a\$10\$CGHjOQPxfP2Ghv |

Sesudah

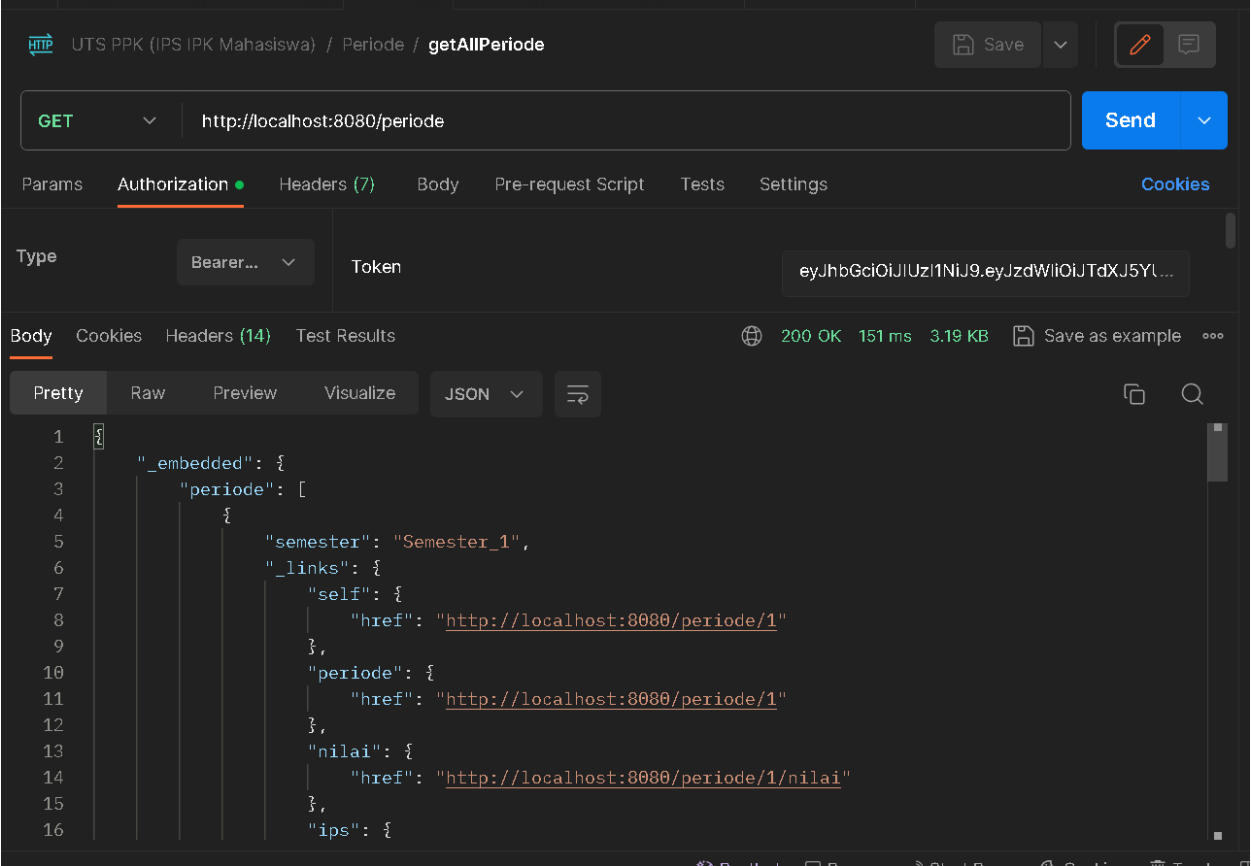




- *GetPeriodeById*



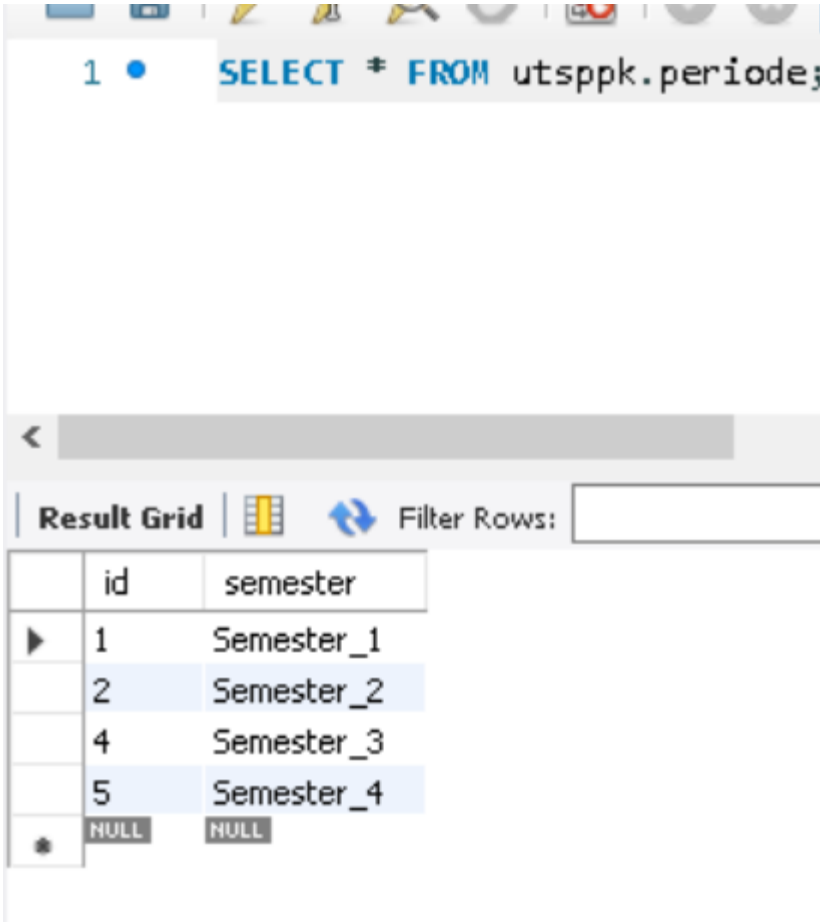
- *GetAllPeriode*



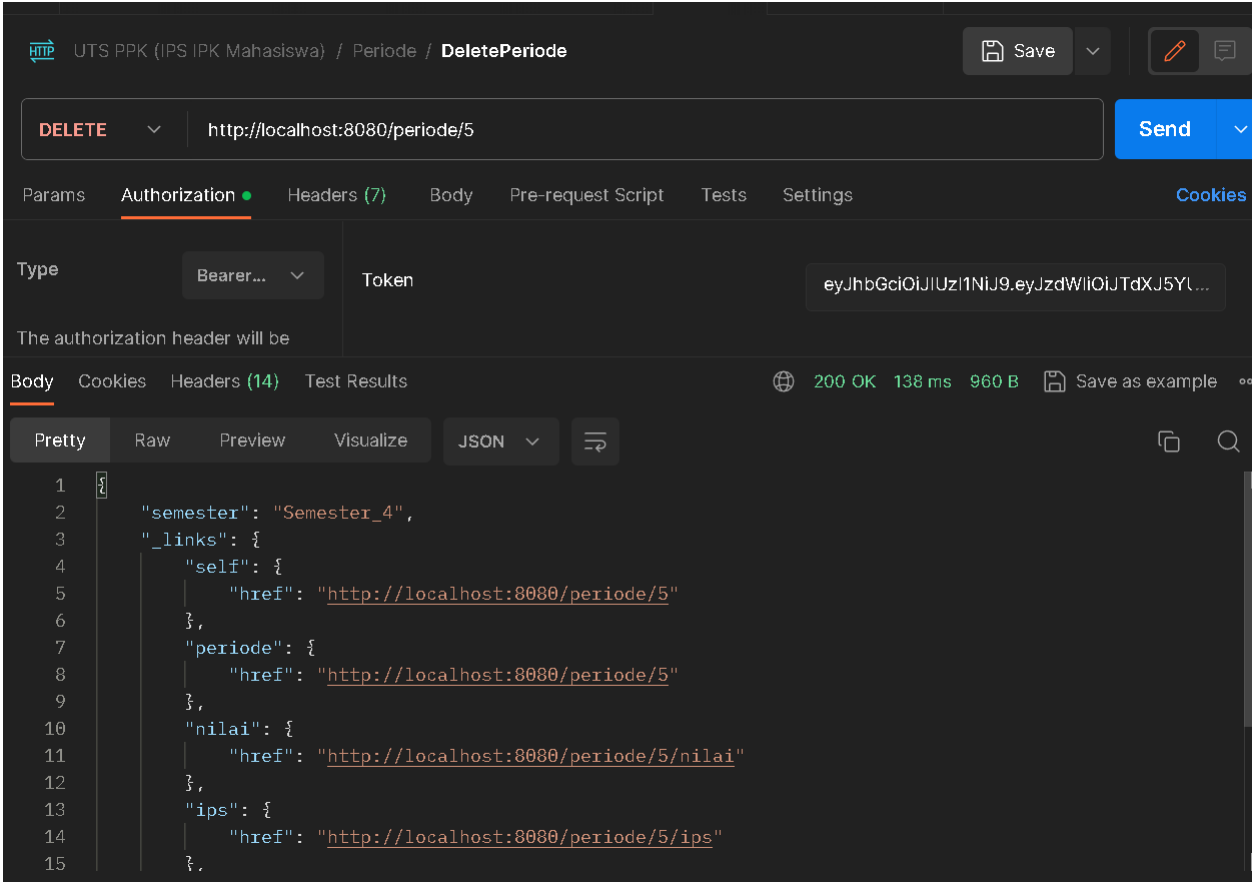
- *DeletePeriodeById*

Pada kasus ini akan diuji delete periode dengan id 5

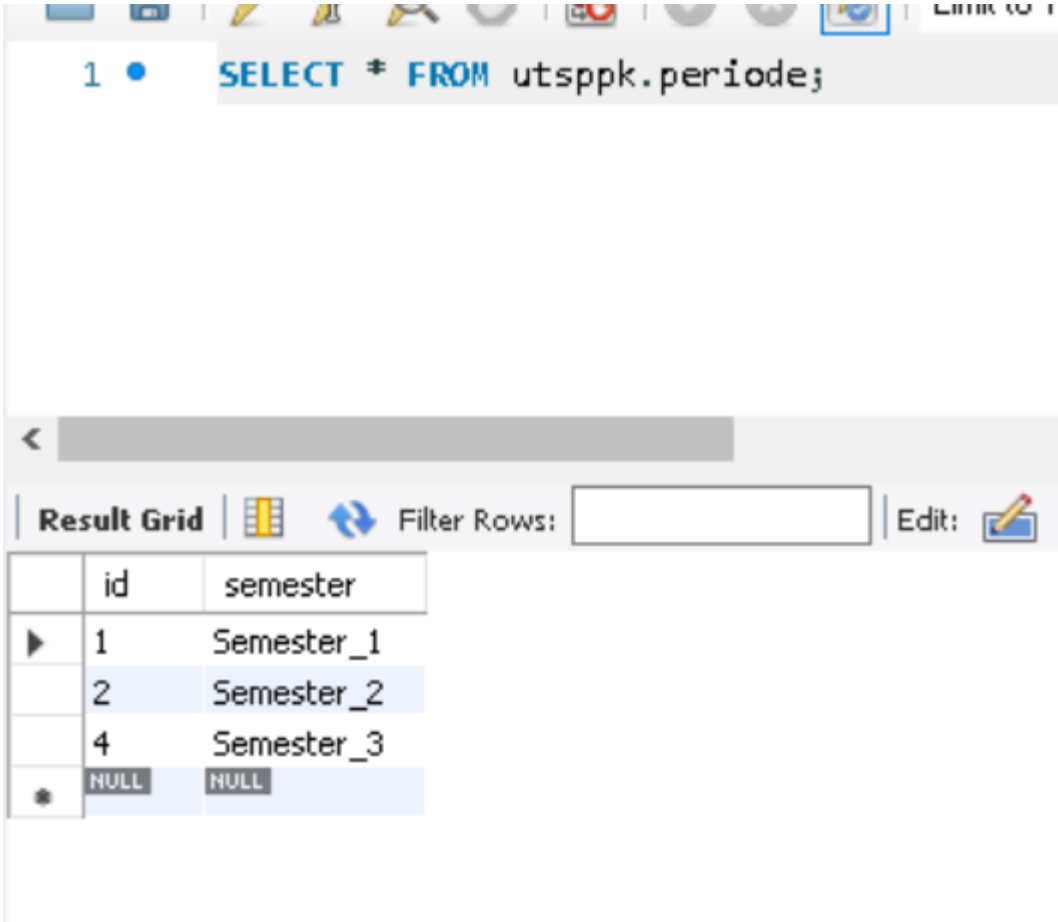
Sebelum



Sesudah



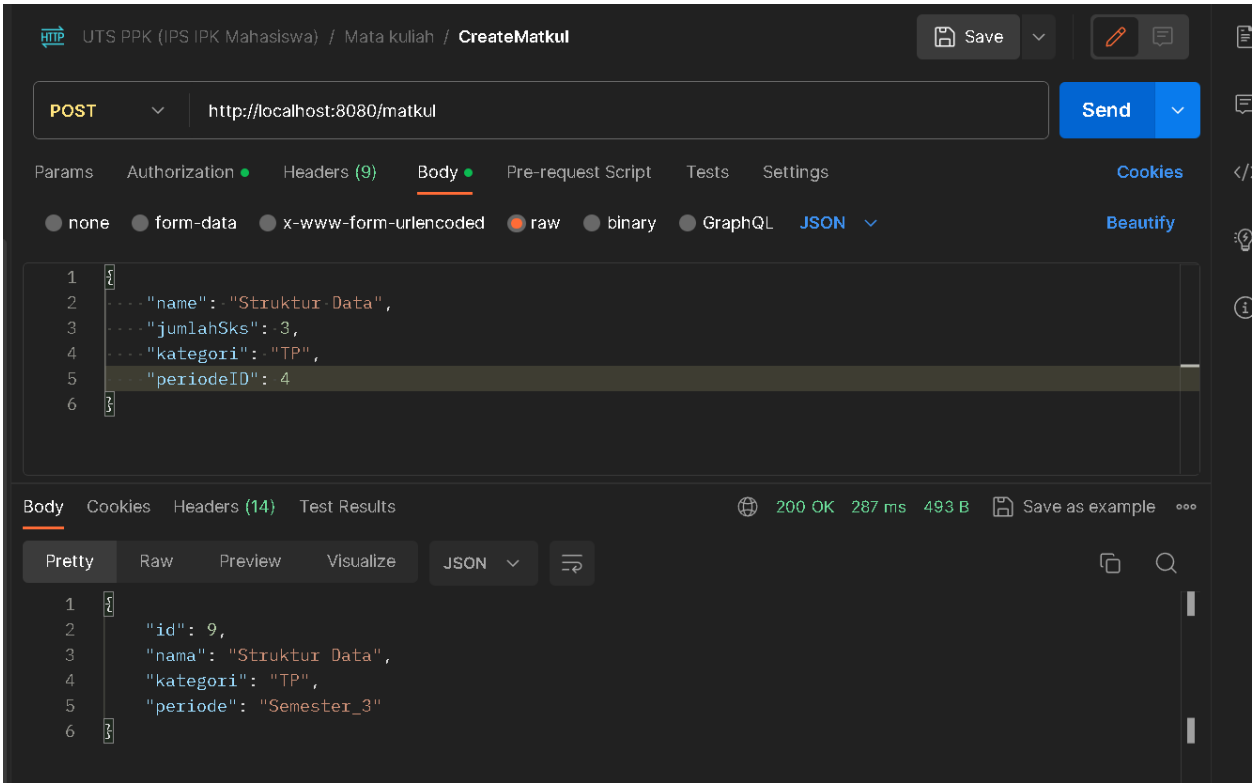




2.5.5 Mata kuliah

- CreateMatkul

Untuk mengakses path ini dan melakukan post diperlukan authority sebagai admin



```
1 • SELECT * FROM utsppek.mata_kuliah;
```

|   | id   | jumlahsks | name                      | periode_id | kategori |
|---|------|-----------|---------------------------|------------|----------|
| ▶ | 1    | 3         | Algoritma dan Pemrograman | 1          | TP       |
|   | 2    | 3         | Analisis Peubah Ganda     | 1          | TP       |
|   | 3    | 2         | Statistika Matematika 1   | 1          | T        |
|   | 5    | 2         | Statistika Matematika 2   | 1          | T        |
|   | 6    | 3         | Kalkulus                  | 2          | T        |
|   | 8    | 3         | ADE                       | 1          | TP       |
|   | 9    | 3         | Struktur Data             | 4          | TP       |
| • | NULL | NULL      | NULL                      | NULL       | NULL     |

- *editMatkul*

request ini juga membutuhkan authority sebagai admin. Pada kasus ini akan dicoba mengubah nama mata kuliah dengan id 8 yang sebelumnya ADE menjadi Analisis Data Kategorik.

Sebelum

|  |   |   |     |   |    |
|--|---|---|-----|---|----|
|  | 8 | 3 | ADE | 1 | TP |
|--|---|---|-----|---|----|

Sesudah

UTS PPK (IPS IPK Mahasiswa) / Mata kuliah / EditMatkul

PATCH

http://localhost:8080/matkul/8

Send

Params

Authorization

Headers (9)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

Beautify

1

{

2

"name": "Analisis Data Kategorik"

3

}

Body

Cookies

Headers (14)

Test Results

200 OK

950 ms

926 B

Save as example

Pretty

Raw

Preview

Visualize

JSON

1

{

2

"name": "Analisis Data Kategorik",

3

"jumlahSKS": 3,

4

"kategori": "TP",

5

"\_links": {

6

"self": {

7

"href": "http://localhost:8080/matkul/8"

8

},

9

"mataKuliah": {

10

"href": "http://localhost:8080/matkul/8"

11

},

12

"periode": {

Postbot

Runner

Start Proxy

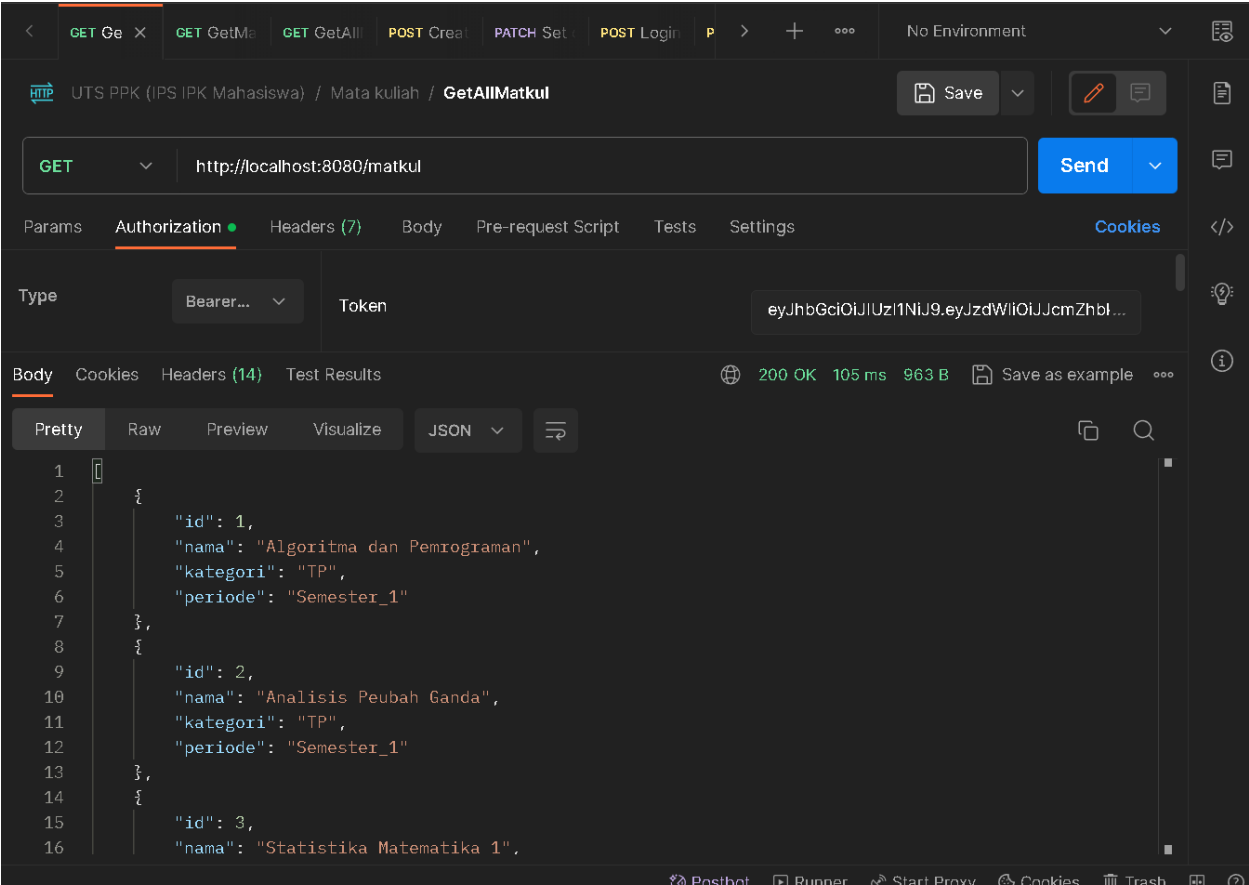
Cookies

Trash

|  |   |   |                         |   |    |
|--|---|---|-------------------------|---|----|
|  | 8 | 3 | Analisis Data Kategorik | 1 | TP |
|--|---|---|-------------------------|---|----|

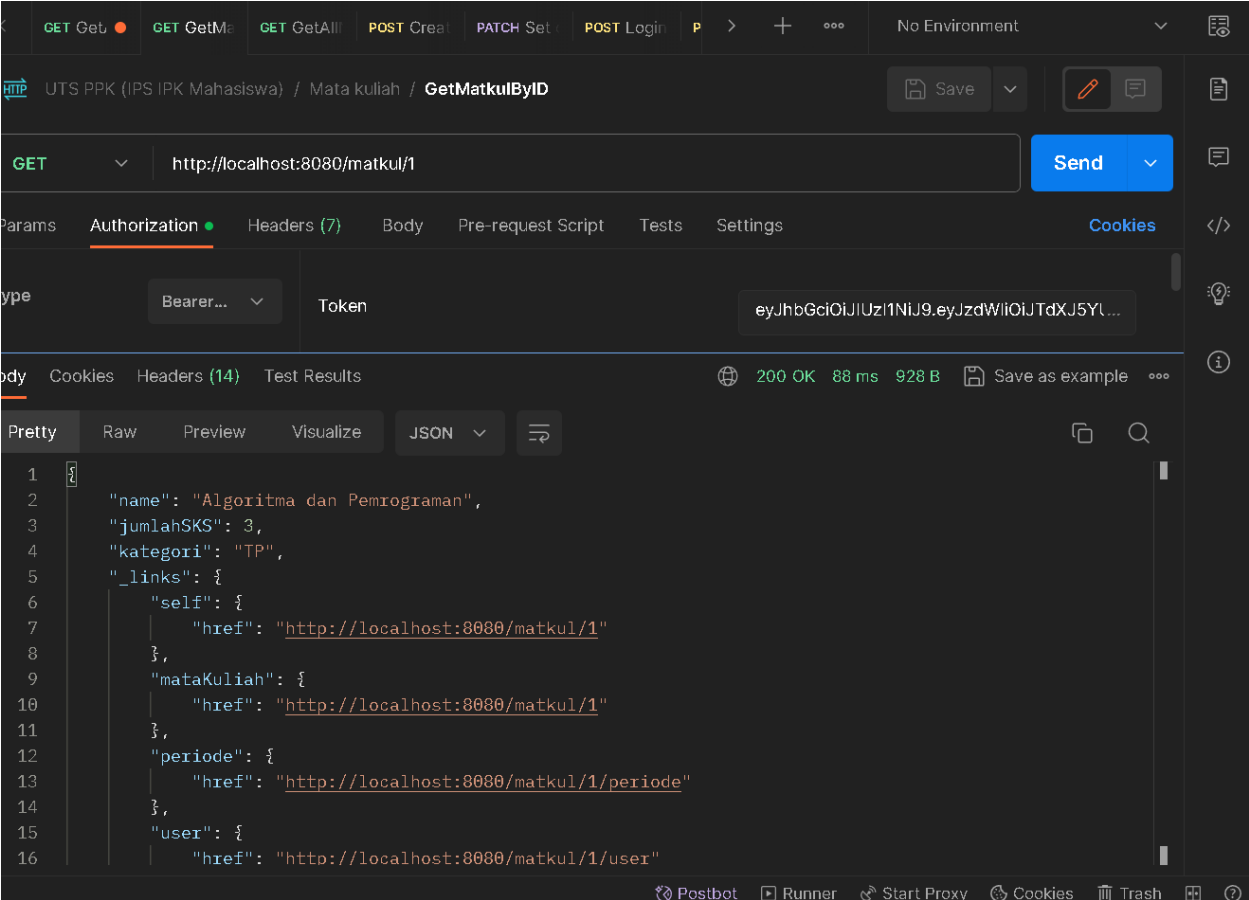
- *getAllMatkul*

request ini dapat dilakukan semua user



- *getMatkulById*

request ini bisa dilakukan oleh user manapun



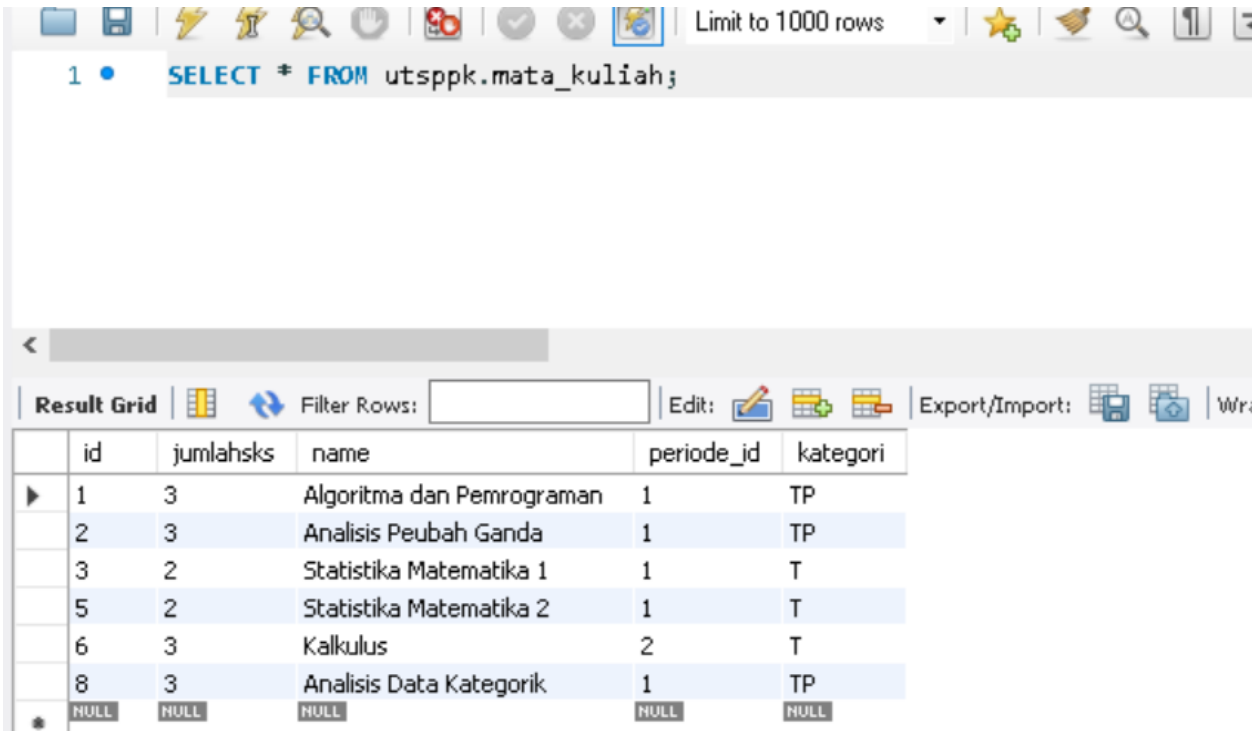
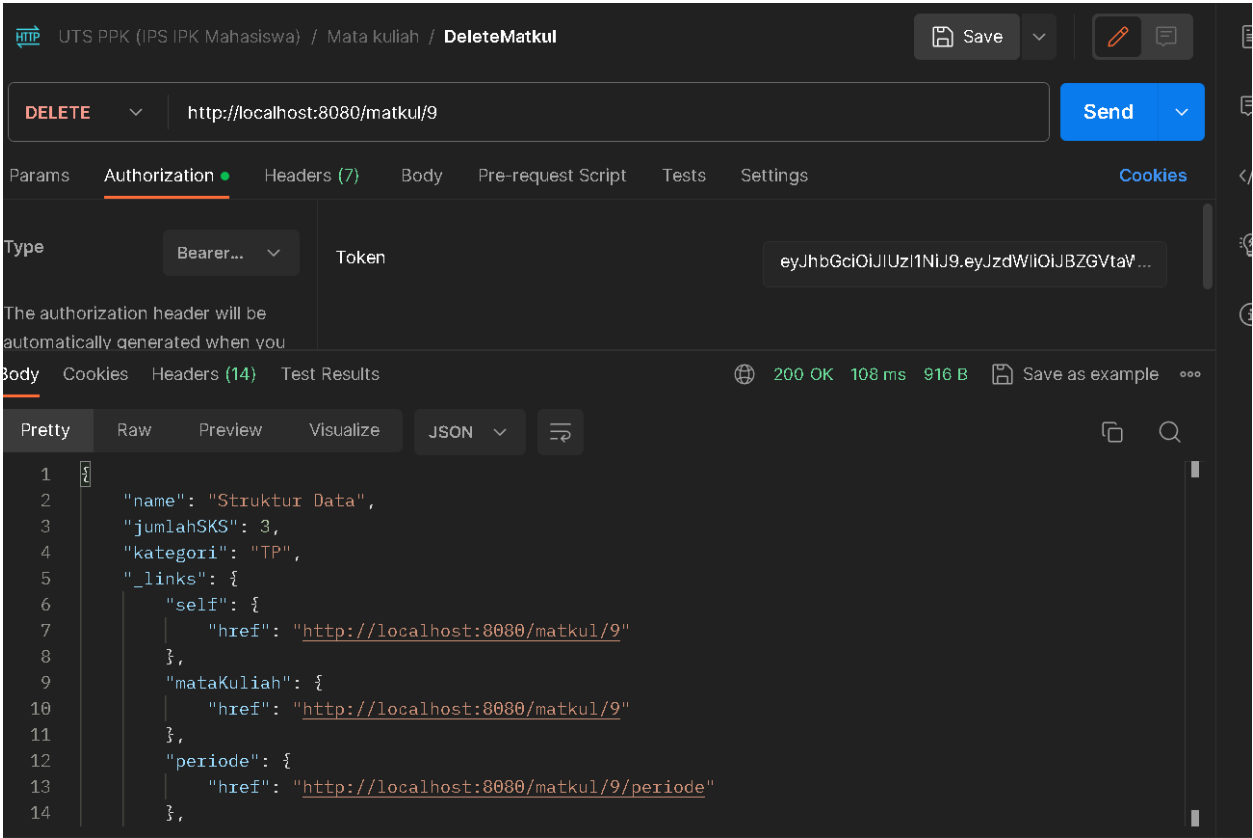
- *DeleteMatkulById*

Request ini hanya dapat diakses oleh admin. Pada kasus ini akan dicoba menghapus data mata kuliah dengan id 9

Sebelum

|  |   |   |               |   |    |
|--|---|---|---------------|---|----|
|  | 9 | 3 | Struktur Data | 4 | TP |
|--|---|---|---------------|---|----|

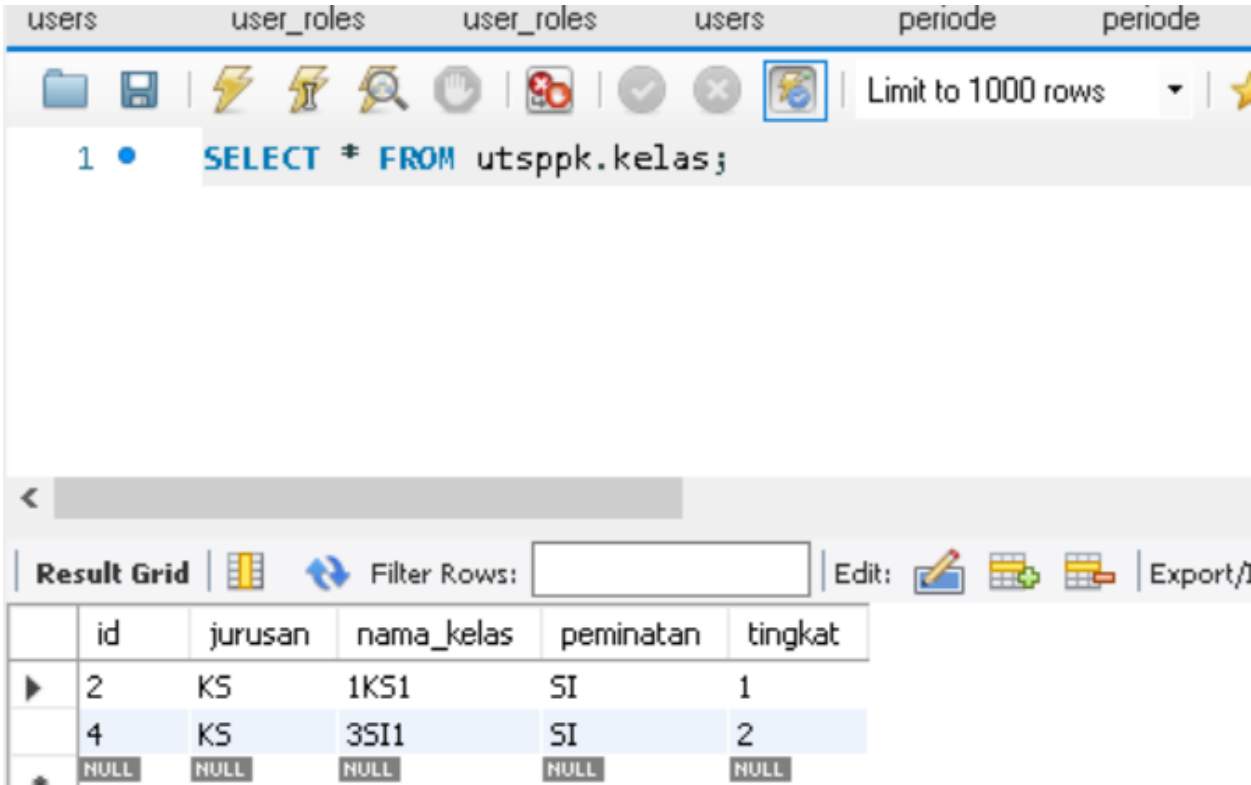
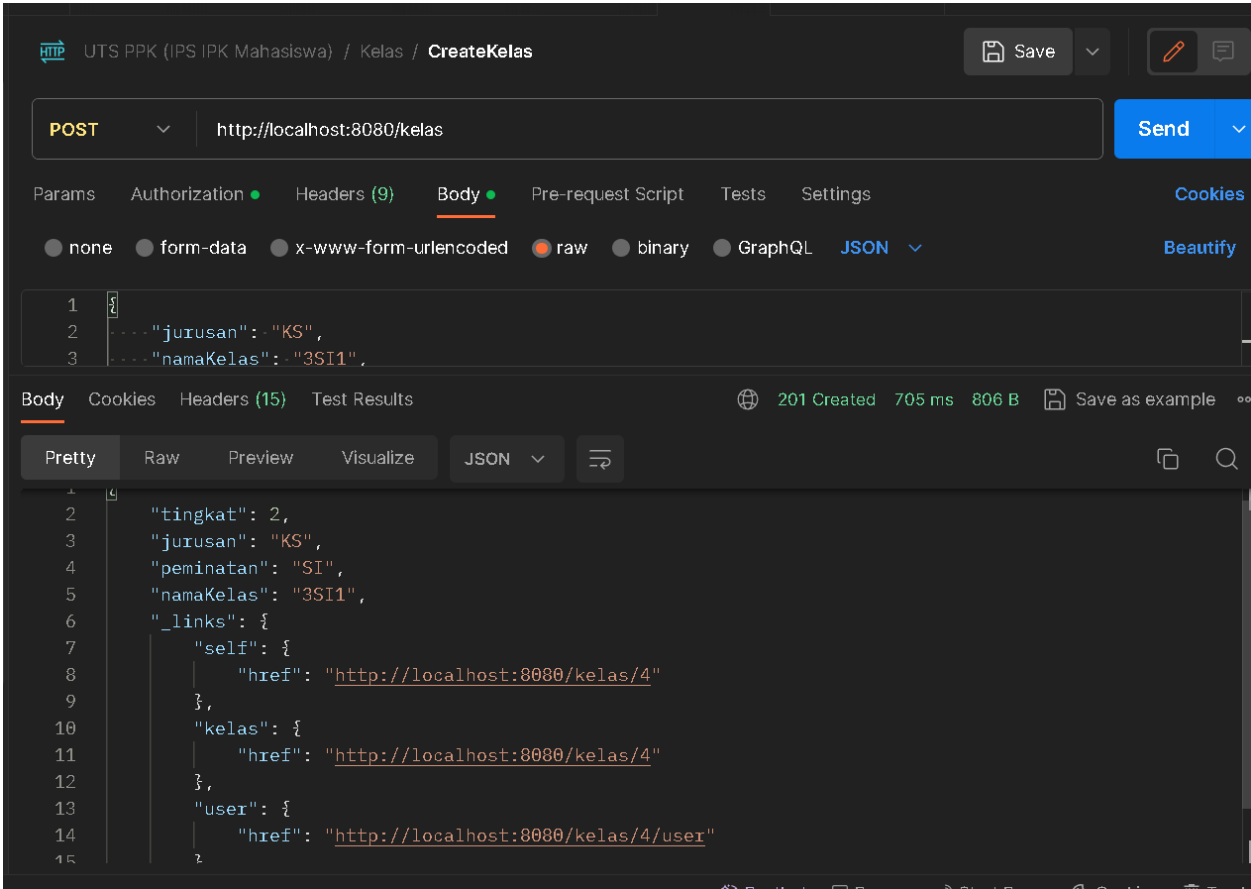
Sesudah



2.5.6 Kelas

- CreateKelas

Request ini hanya dapat dilakukan admin



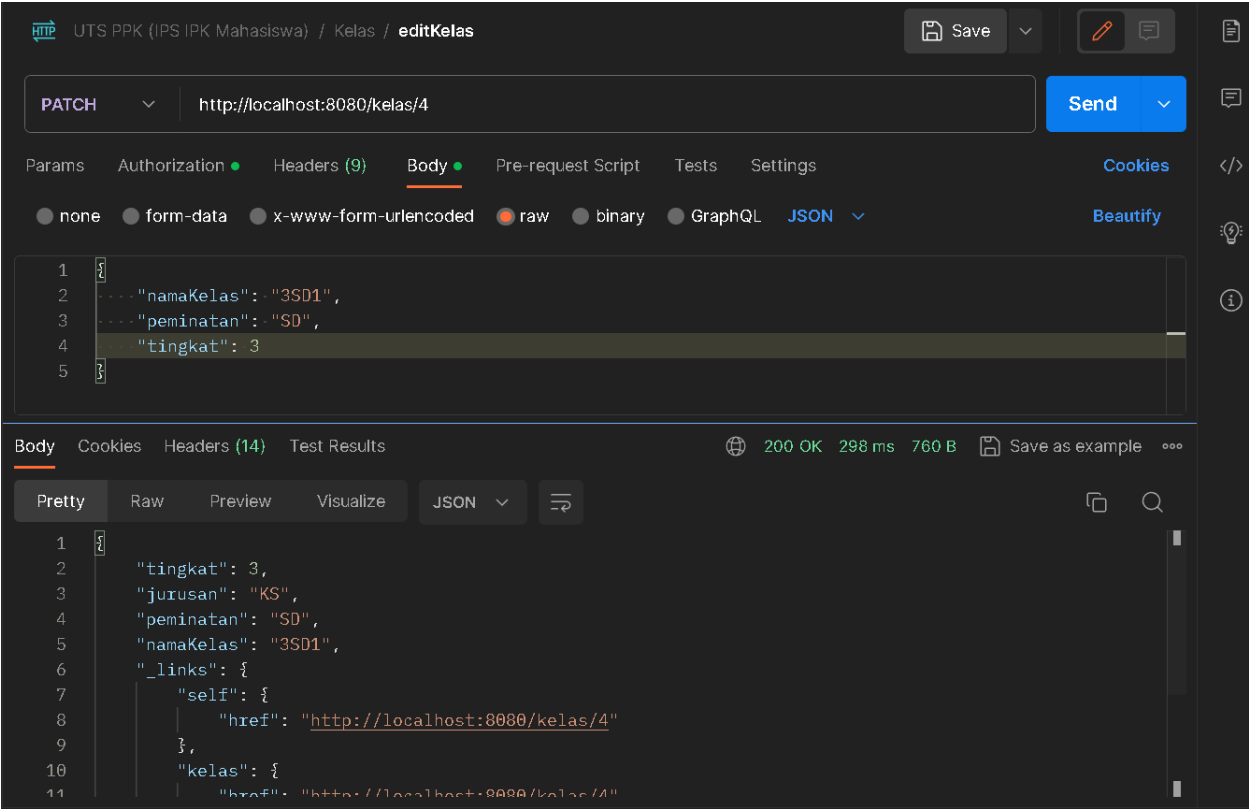
- editKelas

Request ini hanya bisa dilakukan oleh admin. Pada kasus ini akan dicoba mengedit kelas dengan id 4 yang sebelumnya nama kelas nya 3SI1 dan peminatannya SI menjadi SD dan jadi kelas 3SD1 serta membenarkan tingkatnya menjadi tingkat 3

Sebelum

|   |      |      |      |      |      |
|---|------|------|------|------|------|
|   | 4    | KS   | 3SI1 | SI   | 2    |
| ▲ | NULL | NULL | NULL | NULL | NULL |

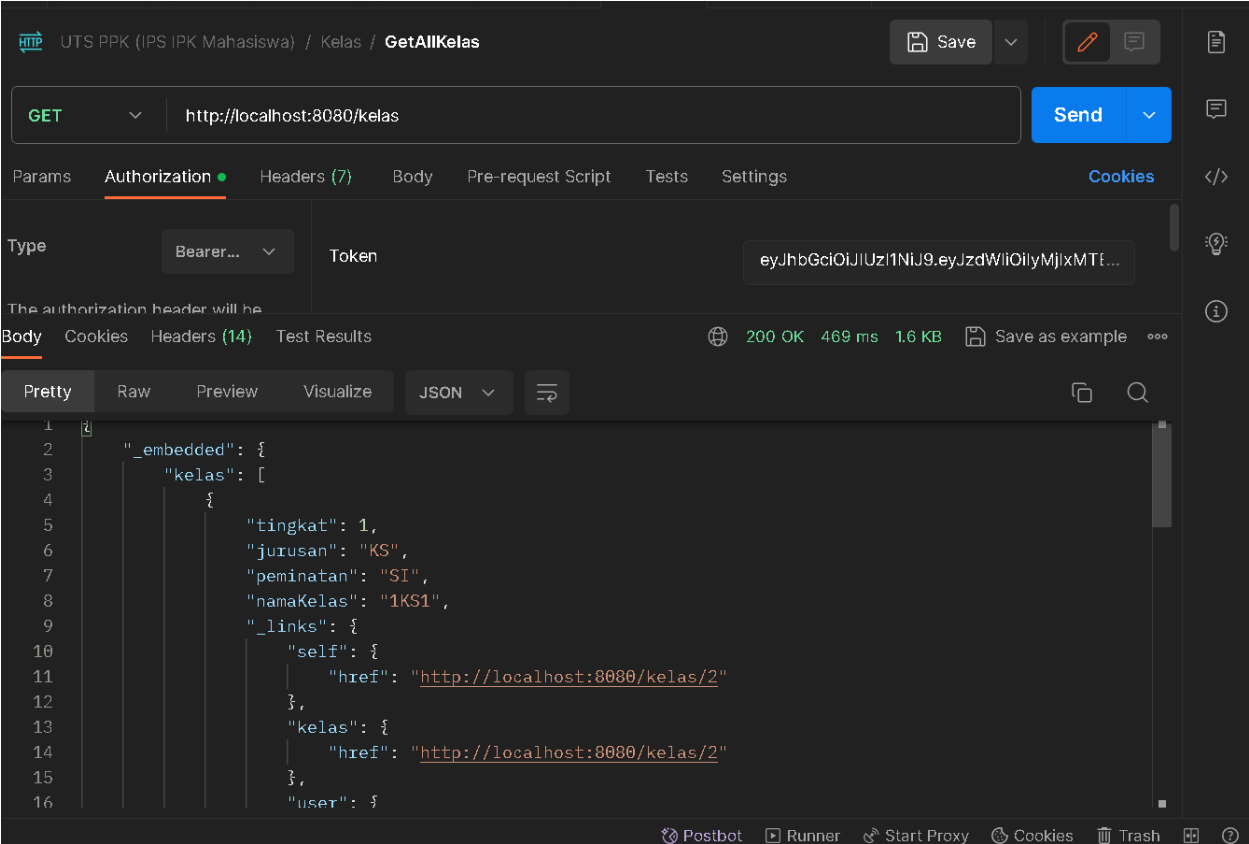
Sesudah



|      |      |      |      |      |
|------|------|------|------|------|
| 4    | KS   | 3SD1 | SD   | 3    |
| NULL | NULL | NULL | NULL | NULL |

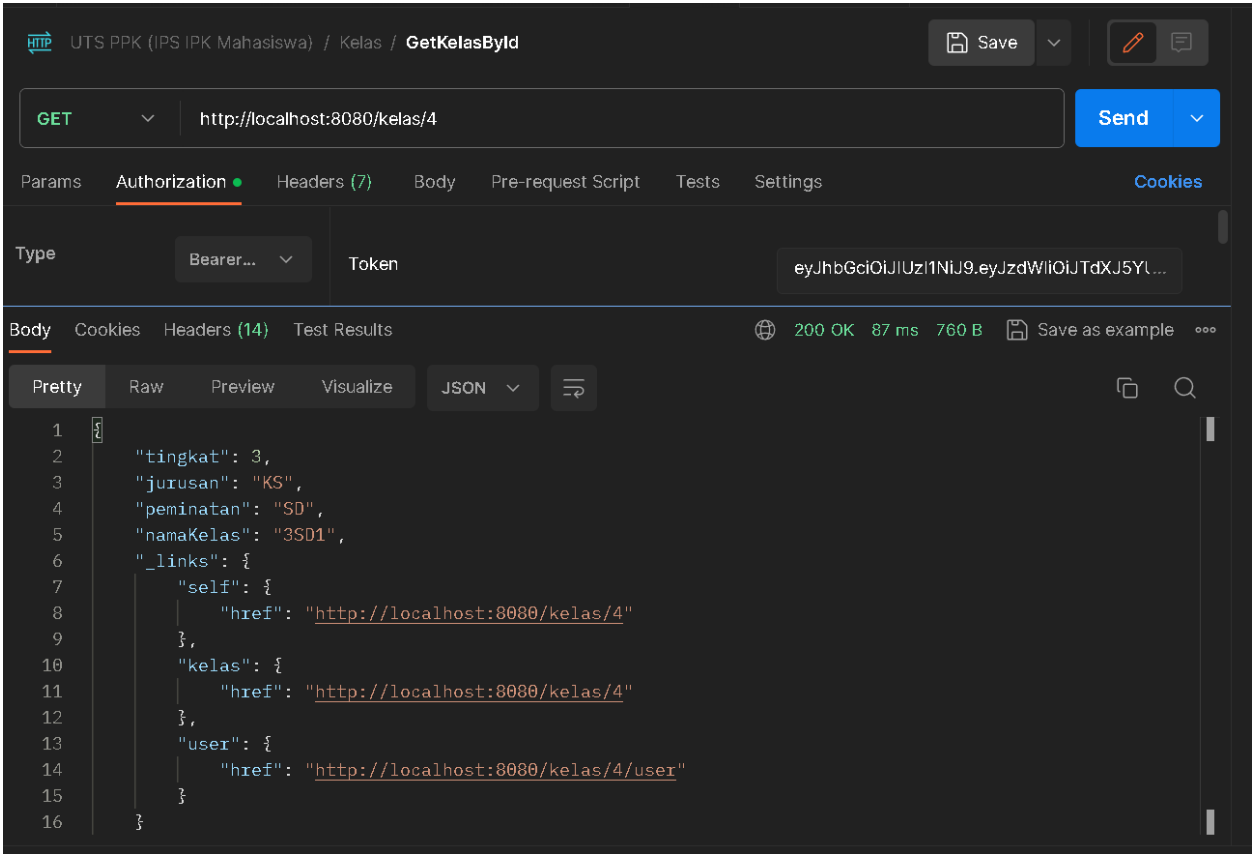
- *GetAllKelas*

Request ini dapat diakses semua role



- *GetKelasById*

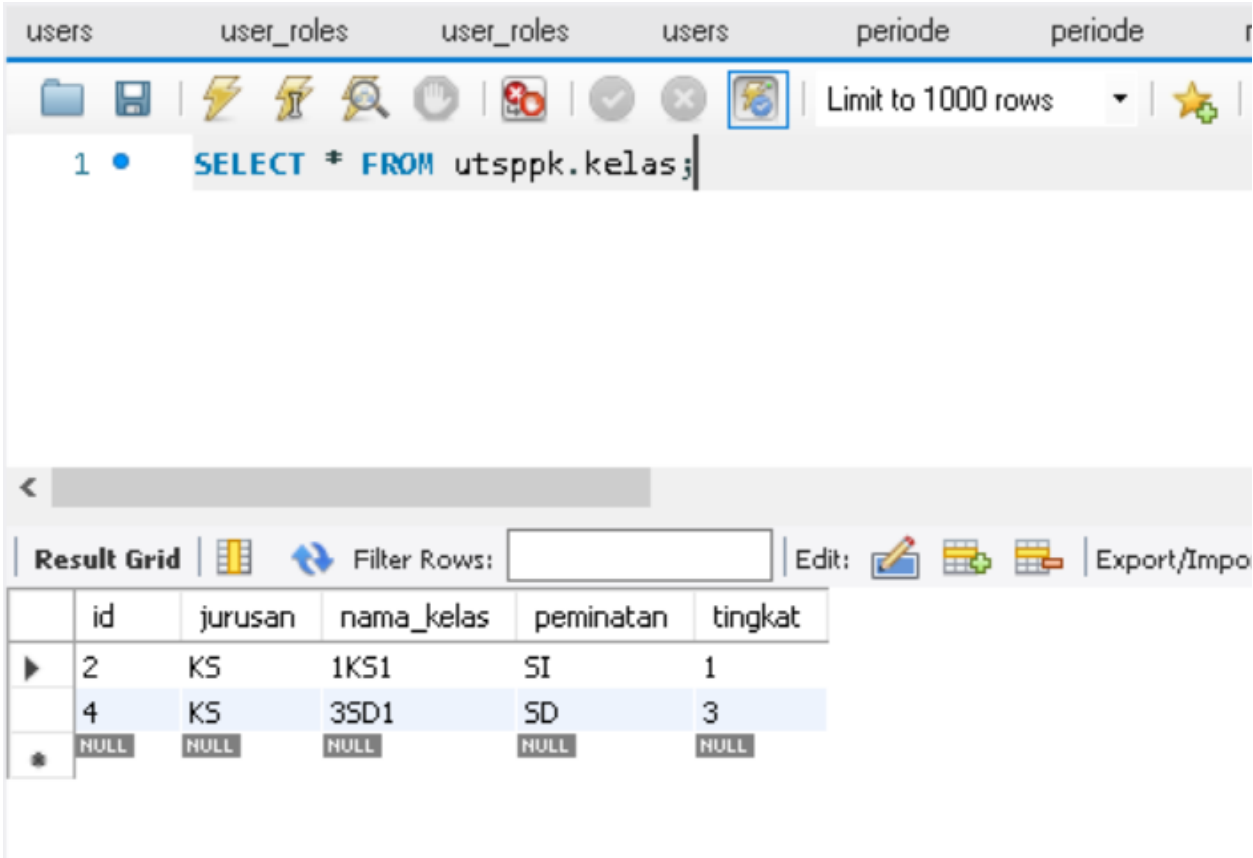
Request ini dapat diakses semua role



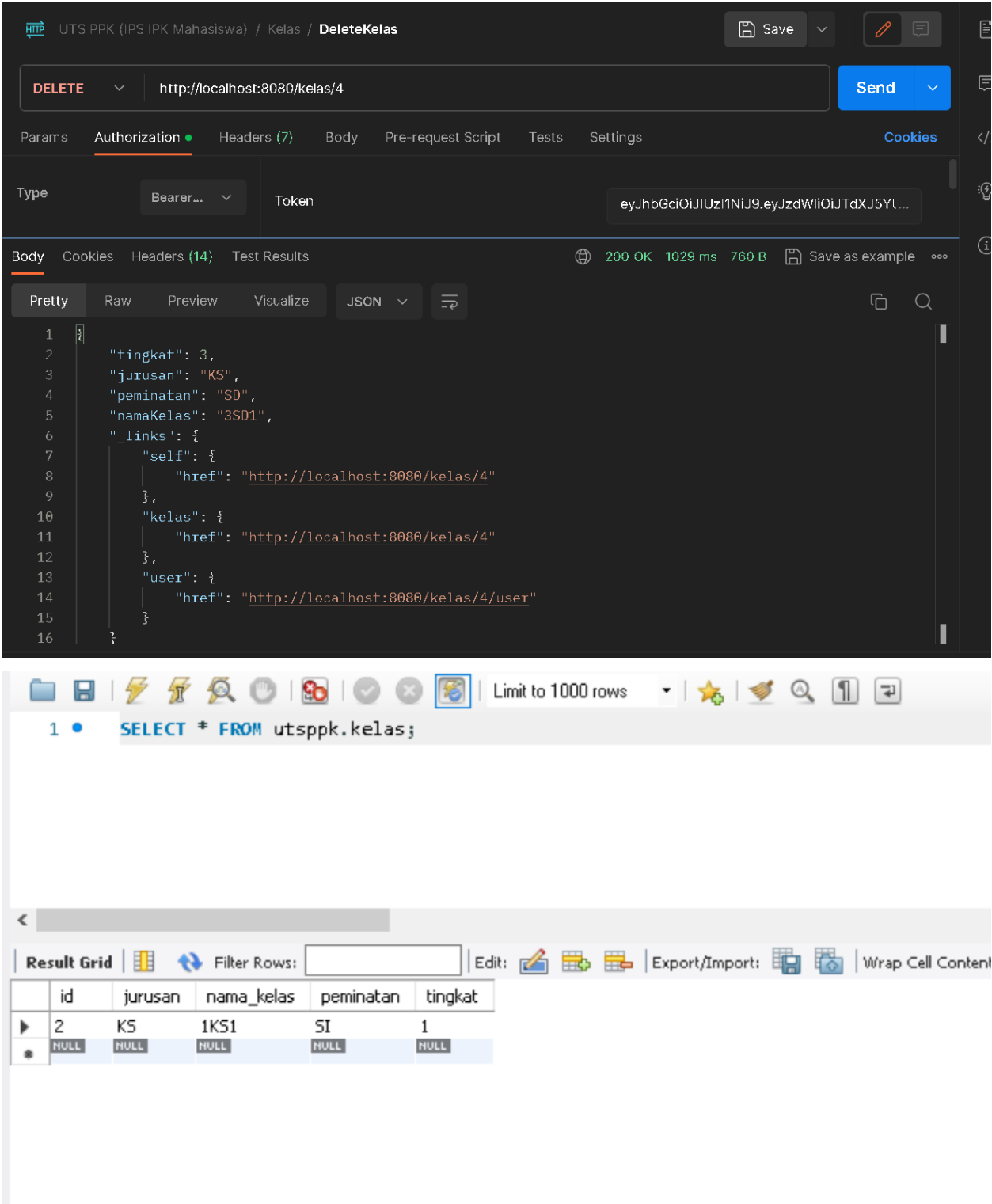
- DeleteKelasById

Request ini hanya dapat dilakukan oleh admin. Pada kasus ini akan dicoba menghapus kelas dengan id 4.

Sebelum



Sesudah

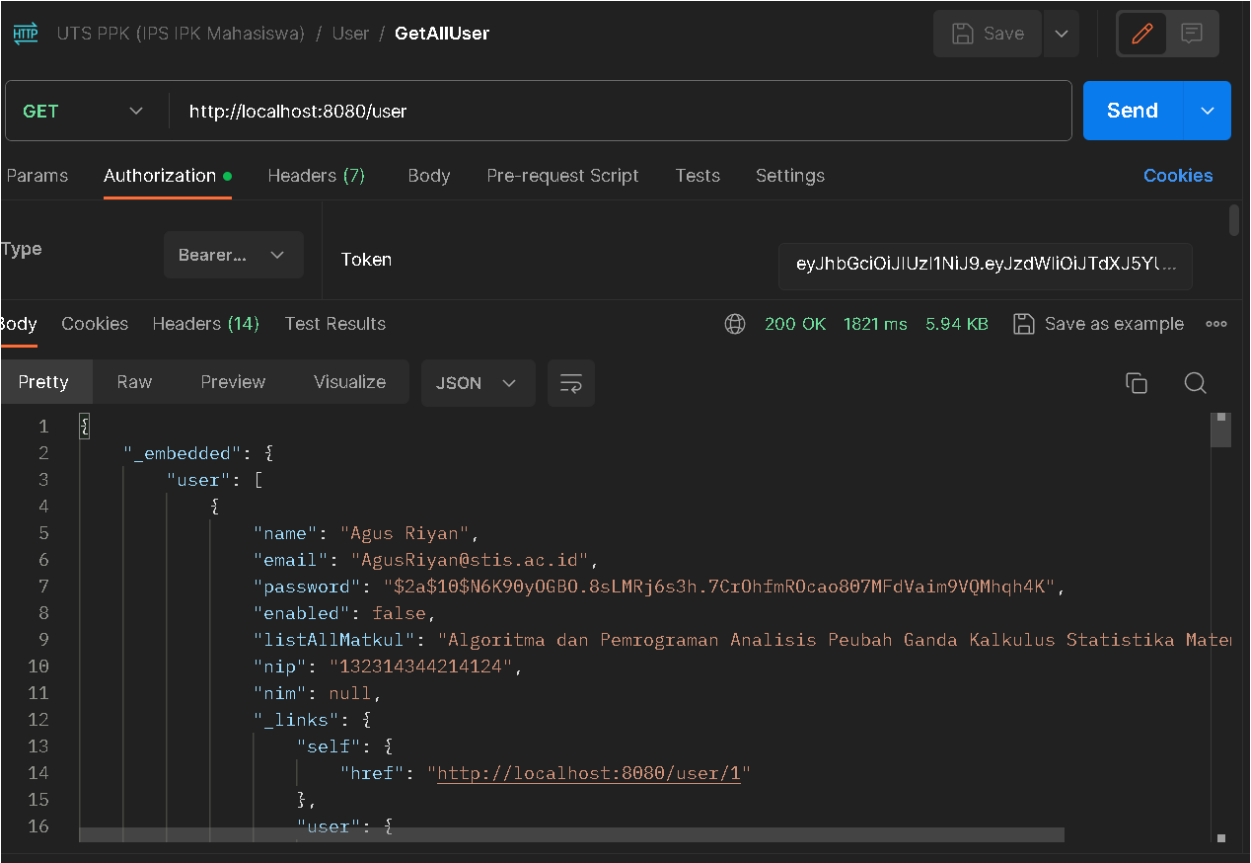


2.5.7 User

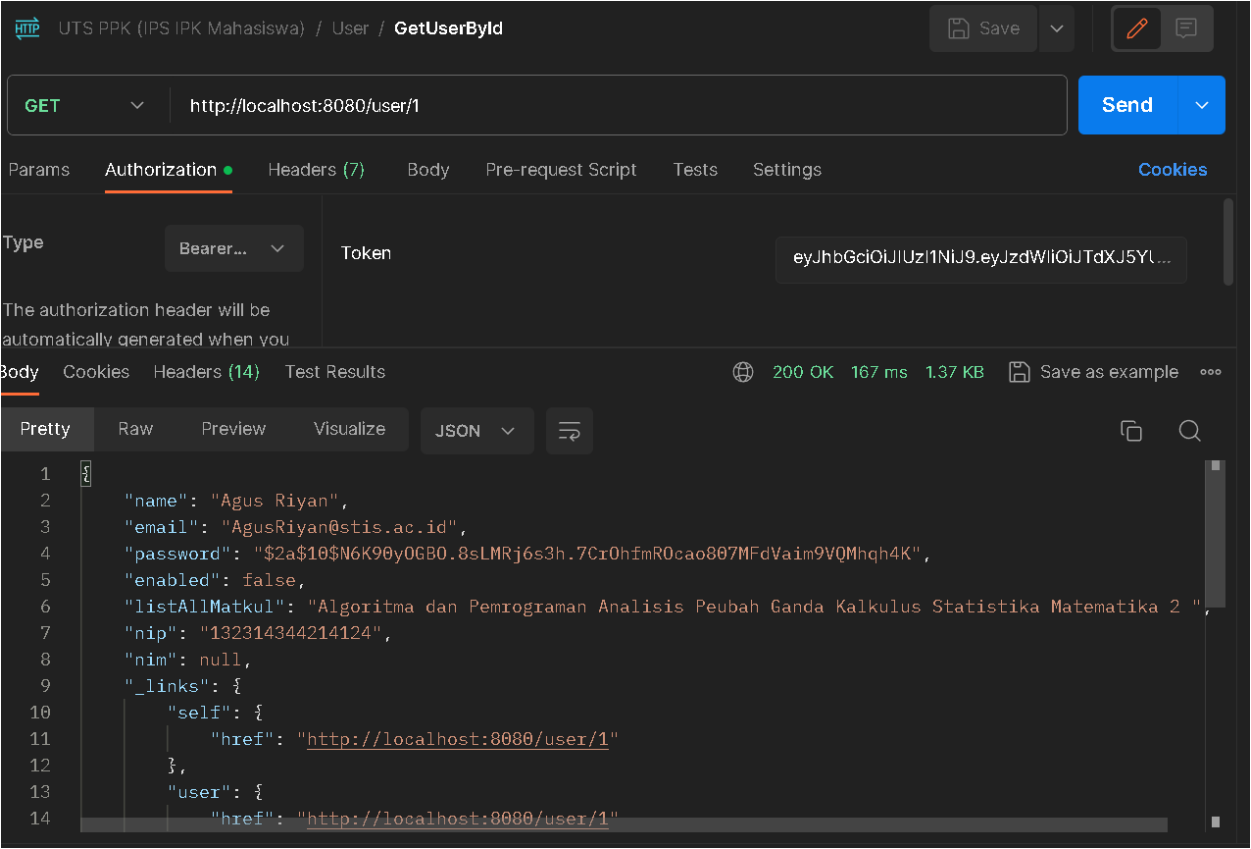
Pada path user ini Sebagian besar request hanya dapat diakses oleh admin. Yang dapat diakses oleh semua role hanya yang methodnya GET.

- *GetAllUser*





- *GetUserById*



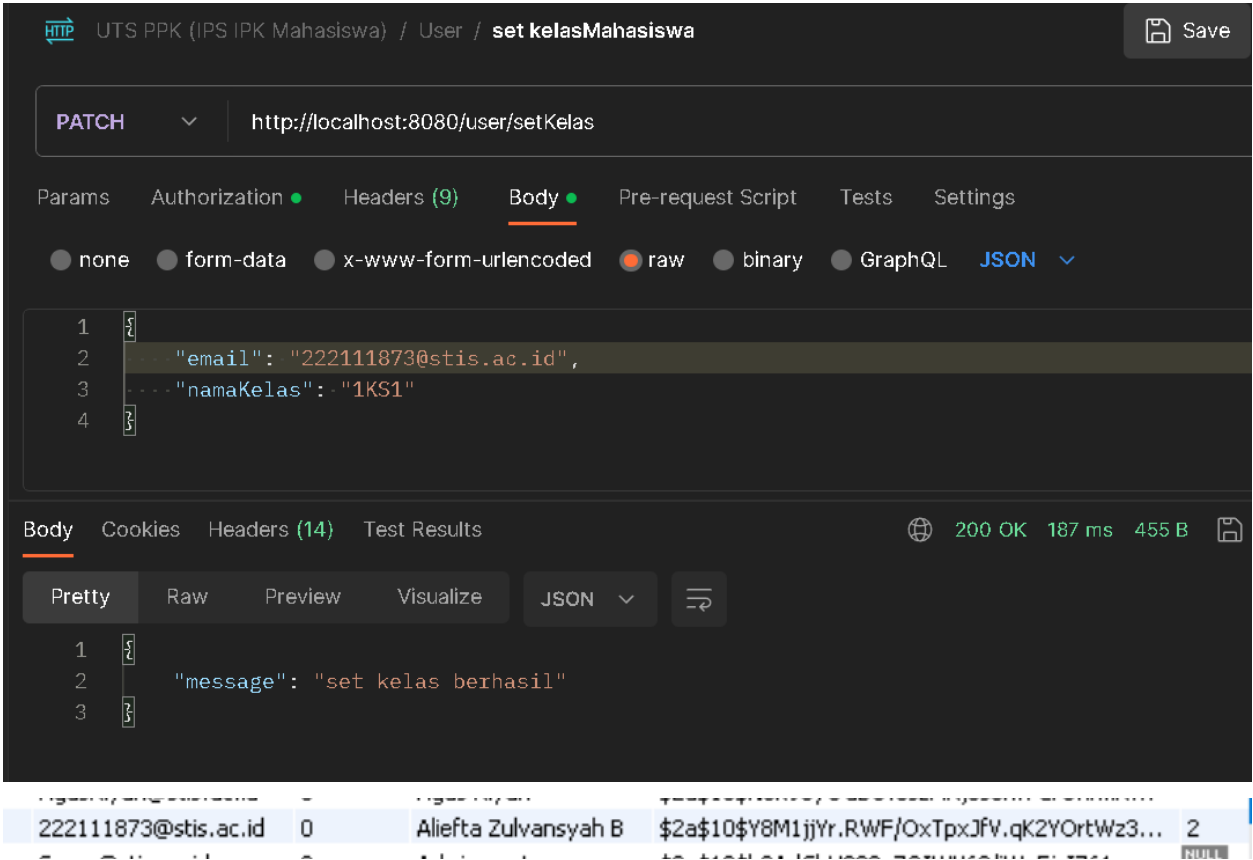
- *DeleteUserById*

Pada kasus ini akan dicoba menghapus user dengan id 7

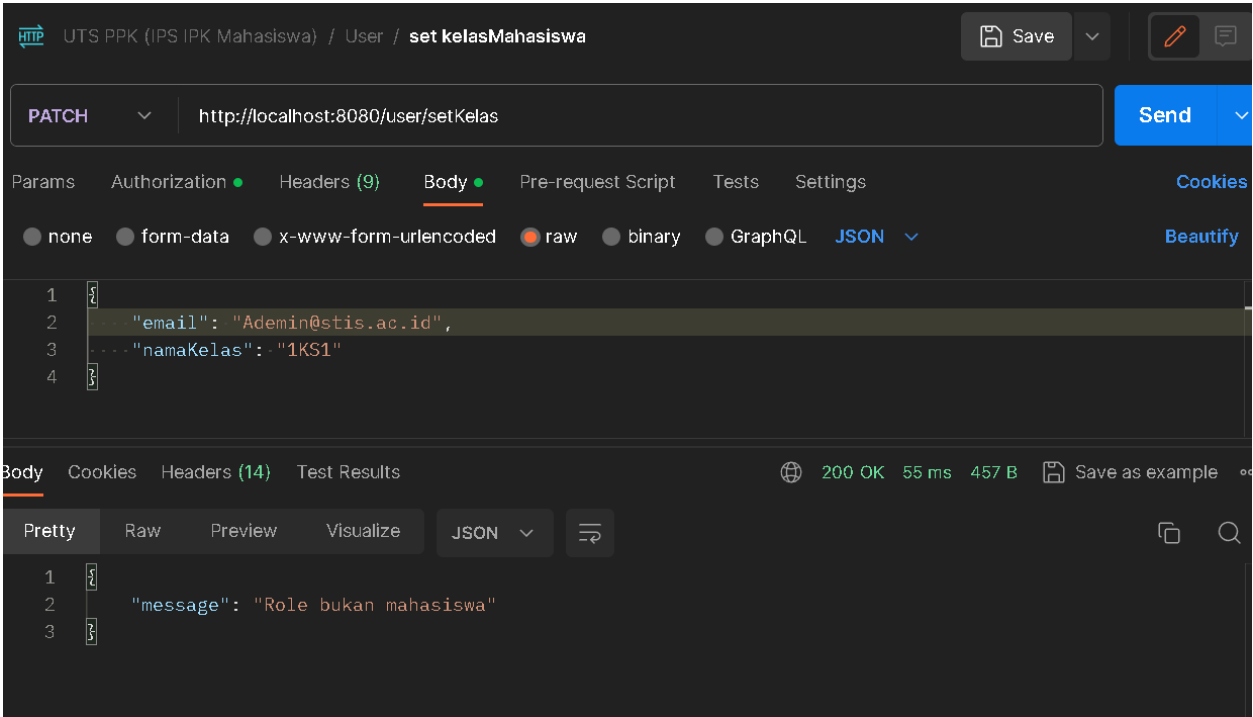
Sebelum



- *SetKelasMahasiswa*



Jika user yang diset kelas bukan mahasiswa maka akan muncul pesan berikut



Lalu set Kelas pun gagal

- *SetMatkulDosen*

Pada kasus ini akan diset user dengan id 5 sebagai dosen pengampu matkul dengan id 2 3 dan 8








Sebelum



periode

mata\_kuliah



kelas



1 • `SELECT * FROM utspk.`

<

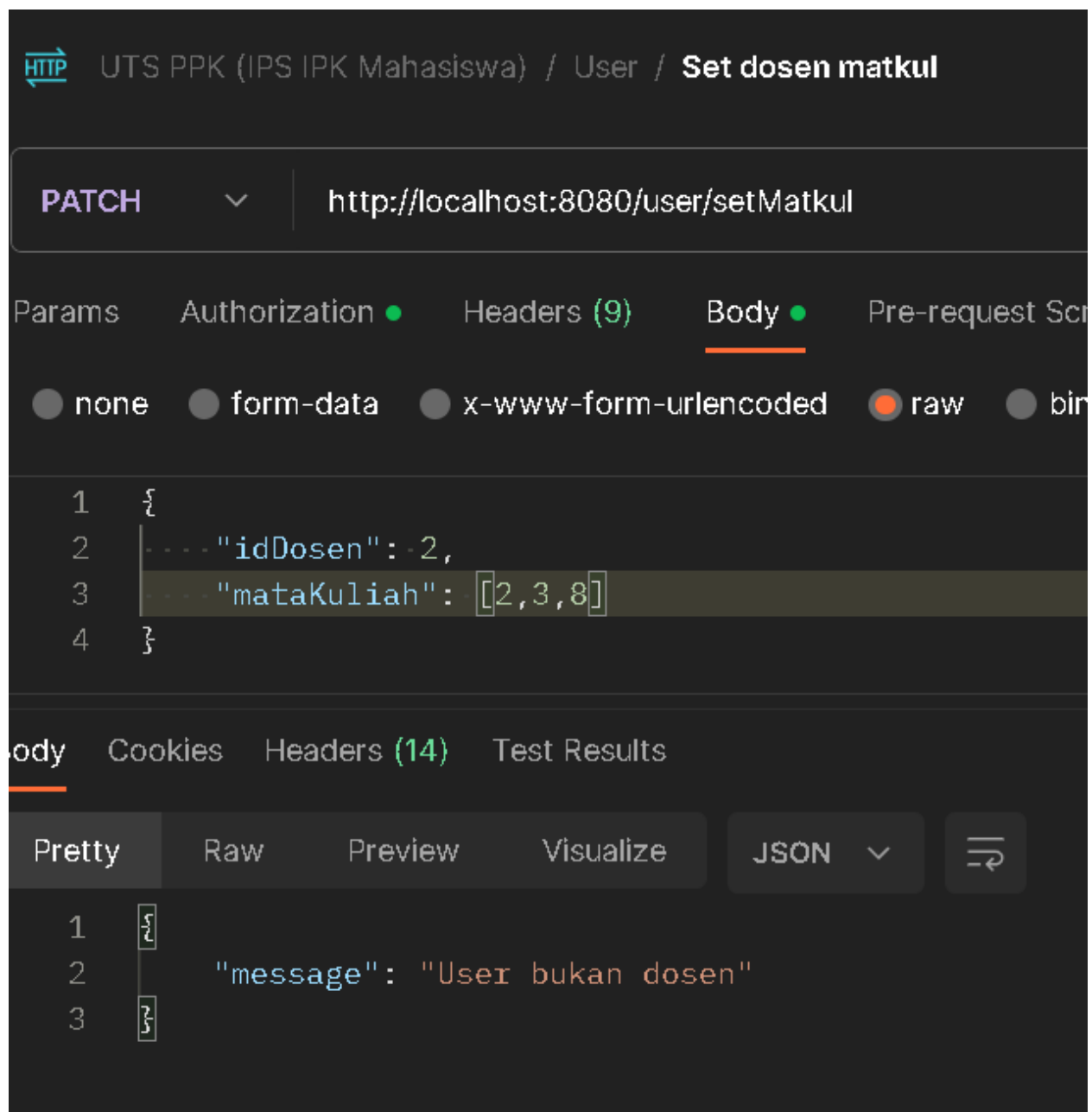
Result Grid

 Filter Rows:

|   | user_id | matkul_id |
|---|---------|-----------|
| ▶ | 1       | 1         |
|   | 1       | 2         |
|   | 1       | 6         |
|   | 1       | 5         |
|   | 5       | 2         |
|   | 5       | 3         |
|   | 5       | 8         |

Refresh data re-exe

Jika user yang diset bukan dosen maka akan muncul pesan berikut



Lalu set kelasnya juga gagal

- *EditUserById*

Pada kasus ini kita akan mengubah data user dengan id 1 dari yang awalnya

|   | id | nim  | nip             | email                | enabled | name       | password                 |
|---|----|------|-----------------|----------------------|---------|------------|--------------------------|
| ▶ | 1  | NULL | 132314344214124 | AgusRiyan@stis.ac.id | 0       | Agus Riyan | \$2a\$10\$N6K90yOGB0.8st |

Menjadi

UTS PPK (IPS IPK Mahasiswa) / User / EditUserById

PATCH

http://localhost:8080/user/1

Send

Params

Authorization

Headers (9)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

Beautify

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

152

153

154

155

156

157

158

159

160

161

162

163

164

165

166

167

168

169

170

171

172

173

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

198

199

200

201

202

203

204

205

206

207

208

209

210

211

212

213

214

215

216

217

218

219

220

221

222

223

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

250

251

252

253

254

255

256

257

258

259

260

261

262

263

264

265

266

267

268

269

270

271

272

273

274

275

276

277

278

279

280

281

282

283

284

285

286

287

288

289

290

291

292

293

294

295

296

297

298

299

300

301

302

303

304

305

306

307

308

309

310

311

312

313

314

315

316

317

318

319

320

321

322

323

324

325

326

327

328

329

330

331

332

333

334

335

336

337

338

339

340

341

342

343

344

345

346

347

348

349

350

351

352

353

354

355

356

357

358

359

360

361

362

363

364

365

366

367

368

369

370

371

372

373

374

375

376

377

378

379

380

381

382

383

384

385

386

387

388

389

390

391

392

393

394

395

396

397

398

399

400

401

402

403

404

405

406

407

408

409

410

411

412

413

414

415

416

417

418

419

420

421

422

423

424

425

426

427

428

429

430

431

432

433

434

435

436

437

438

439

440

441

442

443

444

445

446

447

448

449

450

451

452

453

454

455

456

457

458

459

460

461

462

463

464

465

466

467

468

469

470

471

472

473

474

475

476

477

478

479

480

481

482

483

484

485

486

487

488

489

490

491

492

493

494

495

496

497

498

499

500

501

502

503

504

505

506

507

508

509

510

511

512

513

514

515

516

517

518

519

520

521

522

523

524

525

526

527

528

529

530

531

532

533

534

535

536

537

538

539

540

541

542

543

544

545

546

547

548

549

550

551

552

553

554

555

556

557

558

559

560

561

562

563

564

565

566

567

568

569

570

571

572

573

574

575

576

577

578

579

580

581

582

583

584

585

586

587

588

589

590

591

592

593

594

595

596

597

598

599

600

601

602

603

604

605

606

607

608

609

610

611

612

613

614

615

616

617

618

619

620

621

622

623

624

625

626

627

628

629

630

631

632

633

634

635

636

637

638

639

640

641

642

643

644

645

646

647

648

649

650

651

652

653

654

655

656

657

658

659

660

661

662

663

664

665

666

667

668

669

670

671

672

673

674

675

676

677

678

679

680

681

682

683

684

685

686

687

688

689

690

691

692

693

694

695

696

697

698

699

700

701

702

703

704

705

706

707

708

709

710

711

712

713

714

715

716

717

718

719

720

721

722

723

724

725

726

727

728

729

730

731

732

733

734

735

736

737

738

739

740

741

742

743

744

745

746

747

748

749

750

751

752

753

754

755

756

757

758

759

760

761

762

763

764

765

766

767

768

769

770

771

772

773

774

775

776

777

778

779

780

781

782

783

784

785

786

787

788

789

790

791

792

793

794

795

796

797

798

799

800

801

802

803

804

805

806

807

808

809

810

811

812

813

814

815

816

817

818

819

820

821

822

823

824

825

826

827

828

829

830

831

832

833

834

835

836

837

838

839

840

841

842

843

844

845

846

847

848

849

850

851

852

853

854

855

856

857

858

859

860

861

862

863

864

865

866

867

868

869

870

871

872

873

874

875

876

877

878

879

880

881

882

883

884

885

886

887

888

889

890

891

892

893

894

895

896

897

898

899

900

901

902

903

904

905

906

907

908

909

910

911

912

913

914

915

916

917

918

919

920

921

922

923

924

925

926

927

928

929

930

931

932

933

934

935

936

937

938

939

940

941

942

943

944

945

946

947

948

949

950

951

952

953

954

955

956

957

958

959

960

961

962

963

964

965

966

967

968

969

970

971

972

973

974

975

976

977

978

979

980

981

982

983

984

985

986

987

988

989

990

991

992

993

994

995

996

997

998

999

1000

1001

1002

1003

1004

1005

1006

1007

1008

1009

1010

1011

1012

1013

1014

1015

1016

1017

1018

1019

1020

1021

1022

1023

1024

1025

1026

1027

1028

1029

1030

1031

1032

1033

1034

1035

1036

1037

1038

1039

1040

1041

1042

1043

1044

1045

1046

1047

1048

1049

1050

1051

1052

1053

1054

1055

1056

1057

1058

1059

1060

1061

1062

1063

1064

1065

1066

1067

1068

1069

1070

1071

1072

1073

1074

1075

1076

1077

1078

1079

1080

1081

1082

1083

1084

1085

1086

1087

1088

1089

1090

1091

1092

1093

1094

1095

1096

1097

1098

1099

1100

1101

1102

1103

1104

1105

1106

1107

1108

1109

1110

1111

1112

1113

1114

1115

1116

1117

1118

1119

1120

1121

1122

1123

1124

1125

1126

1127

1128

1129

1130

1131

1132

1133

1134

1135

1136

1137

1138

1139

1140

1141

1142

1143

1144

1145

1146

1147

1148

1149

1150

1151

1152

1153

1154

1155

1156

1157

1158

1159

1160

1161

1162

1163

1164

1165

1166

1167

1168

1169

1170

1171

1172

1173

1174

1175

1176

1177

1178

1179

1180

1181

1182

1183

1184

1185

1186

1187

1188

1189

1190

1191

1192

1193

1194

1195

1196

1197

1198

1199

1200

1201

1202

1203

1204

1205

1206

1207

1208

1209

1210

1211

1212

1213

1214

1215

1216

1217

1218

1219

1220

1221

1222

1223

1224

1225

1226

1227

1228

1229

1230

1231

1232

1233

1234

1235

1236

1237

1238

1239

1240

1241

1242

1243

1244

1245

1246

1247

1248

1249

1250

1251

1252

1253

1254

1255

1256

1257

1258

1259

1260

1261

1262

1263

1264

1265

1266

1267

1268

1269

1270

1271

1272

1273

1274

1275

1276

1277

1278

1279

1280

1281

1282

1283

1284

1285

1286

1287

1288

1289

1290

1291

1292

1293

1294

1295

1296

1297

1298

1299

1300

1301

1302

1303

1304

1305

1306

1307

1308

1309

1310

1311

1312

1313

1314

1315

1316

1317

1318

1319

1320

1321

1322

1323

1324

1325

1326

1327

1328

1329

1330

1331

1332

1333

1334

1335

1336

1337

1338

1339

1340

1341

1342

1343

1344

1345

1346

1347

1348

1349

1350

1351

1352

1353

1354

1355

1356

1357

1358

1359

1360

1361

1362

1363

1364

1365

1366

1367

1368

1369

1370

1371

1372

1373

1374

1375

1376

1377

1378

1379

1380

1381

1382

1383

1384

1385

1386

1387

1388

1389

1390

1391

1392

1393

1394

1395

1396

1397

1398

1399

1400

1401

1402

1403

1404

1405

1406

1407

1408

1409

1410

1411

1412

1413

1414

1415

1416

1417

1418

1419

1420

1421

1422

1423

1424

1425

1426

1427

1428

1429

1430

1431

1432

1433

1434

1435

1436

1437

1438

1439

1440

1441

1442

1443

1444

1445

1446

1447

1448

1449

1450

1451

1452

1453

1454

1455

1456

1457

1458

1459

1460

1461

1462

1463

1464

1465

1466

1467

1468

1469

1470

1471

1472

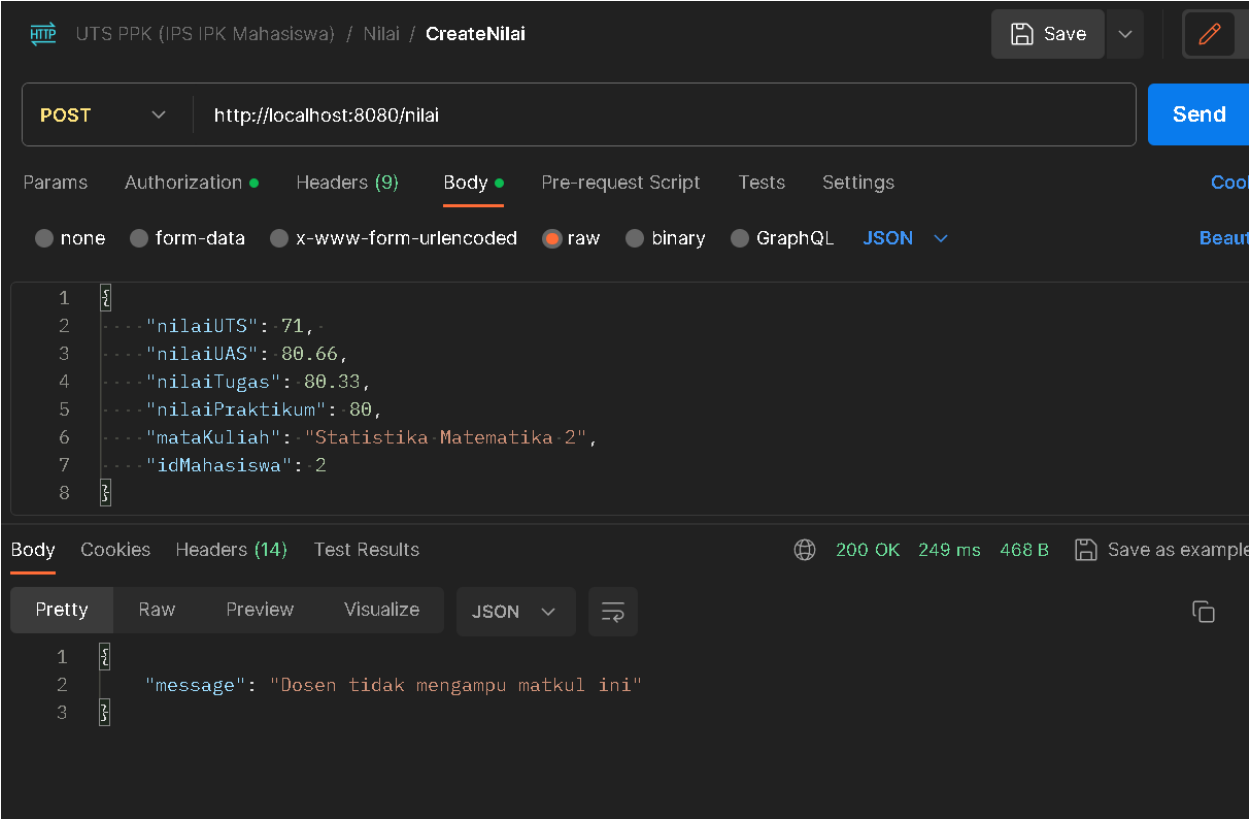
1473

1474

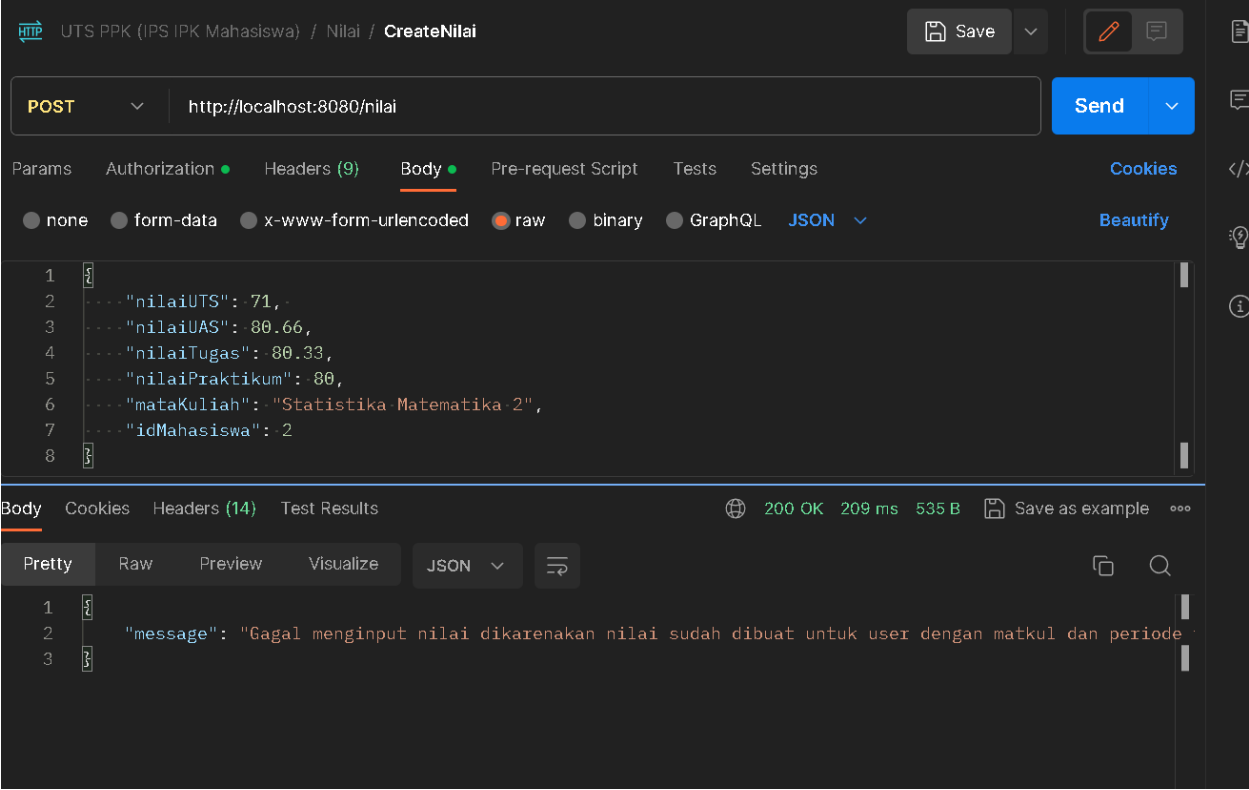
1475

1476

1477



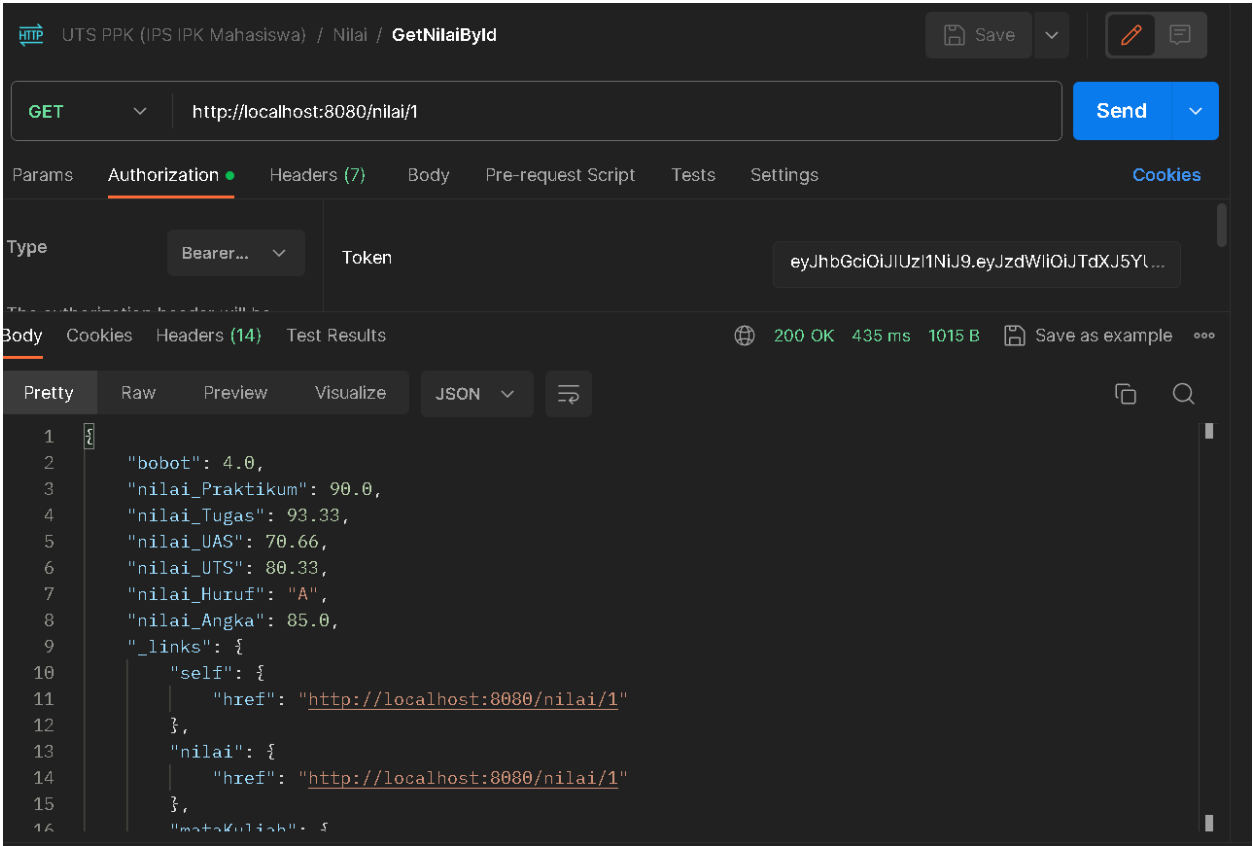
Jika nilai dengan matkul tersebut sudah pernah diset pada user



- *GetNilaiById*

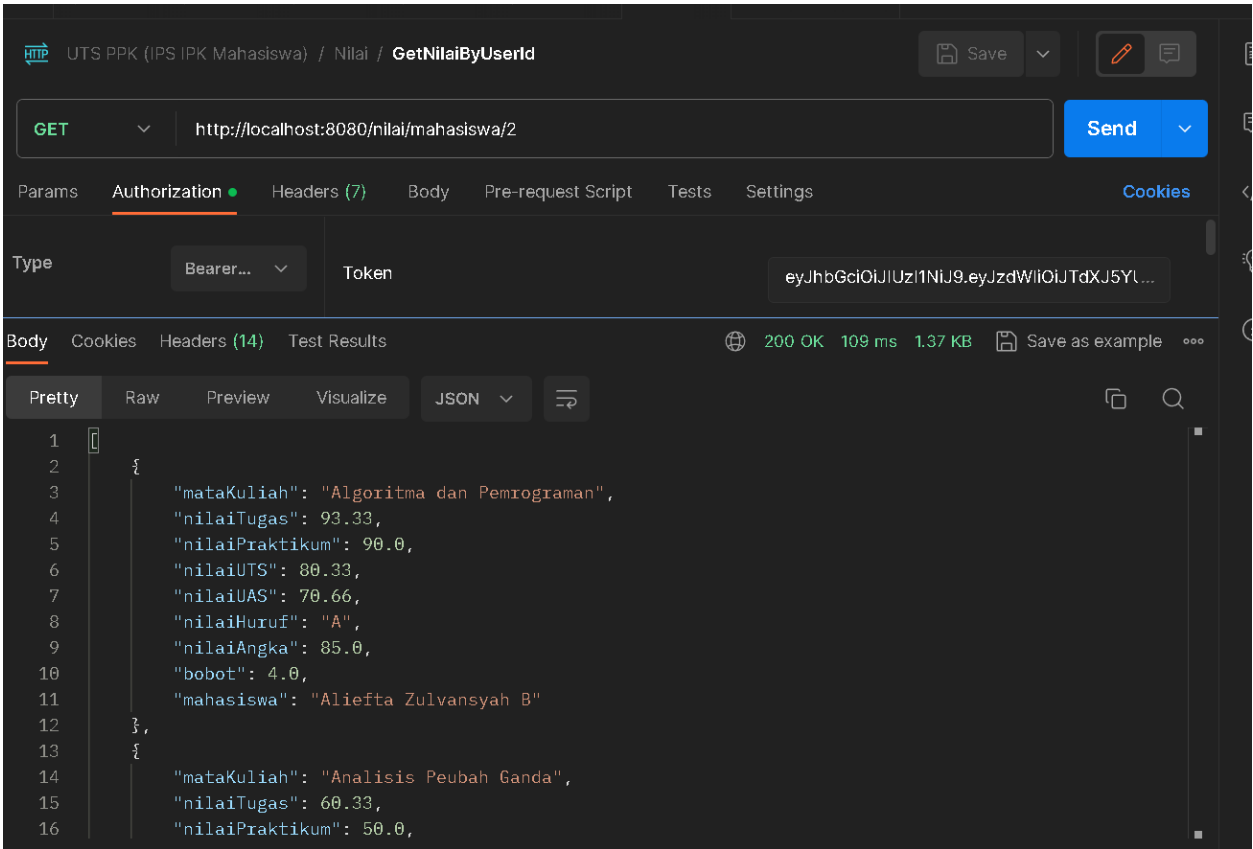
Hanya bisa diakses role dosen dan admin



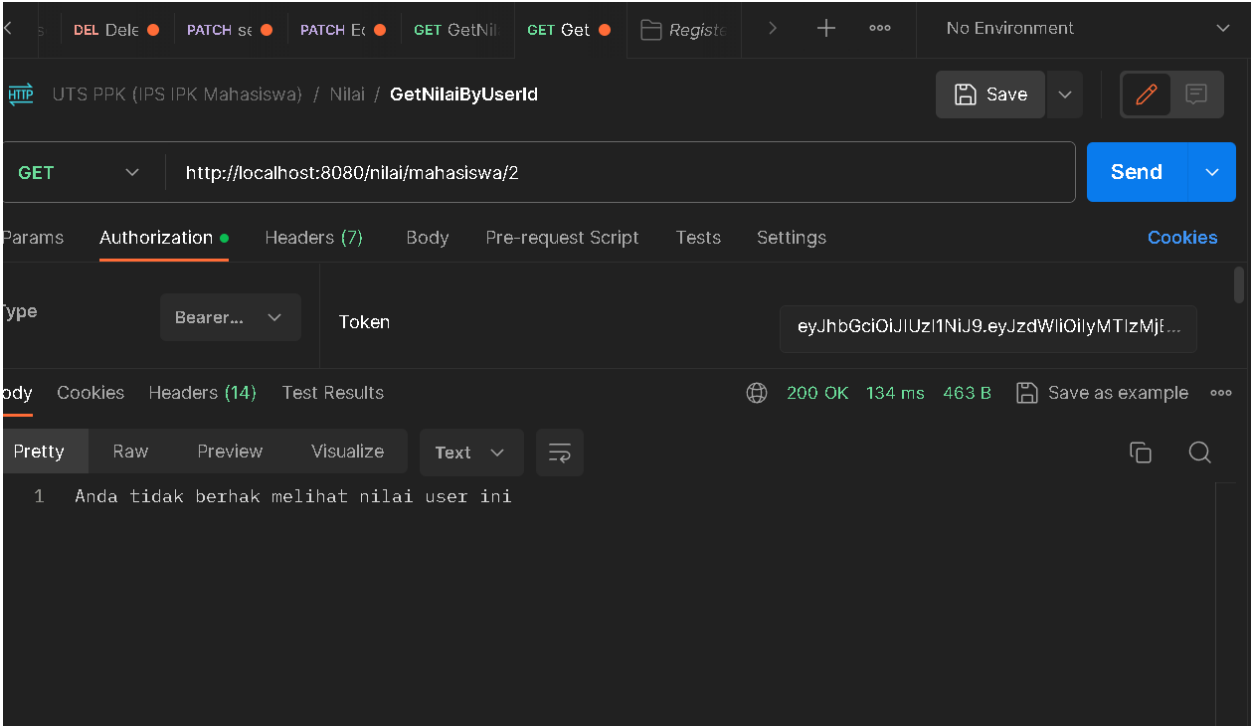


- *GetNilaiByUserId*

Dapat diakses semua role. Namun jika mahasiswa hanya bisa mengecek miliknya sendiri



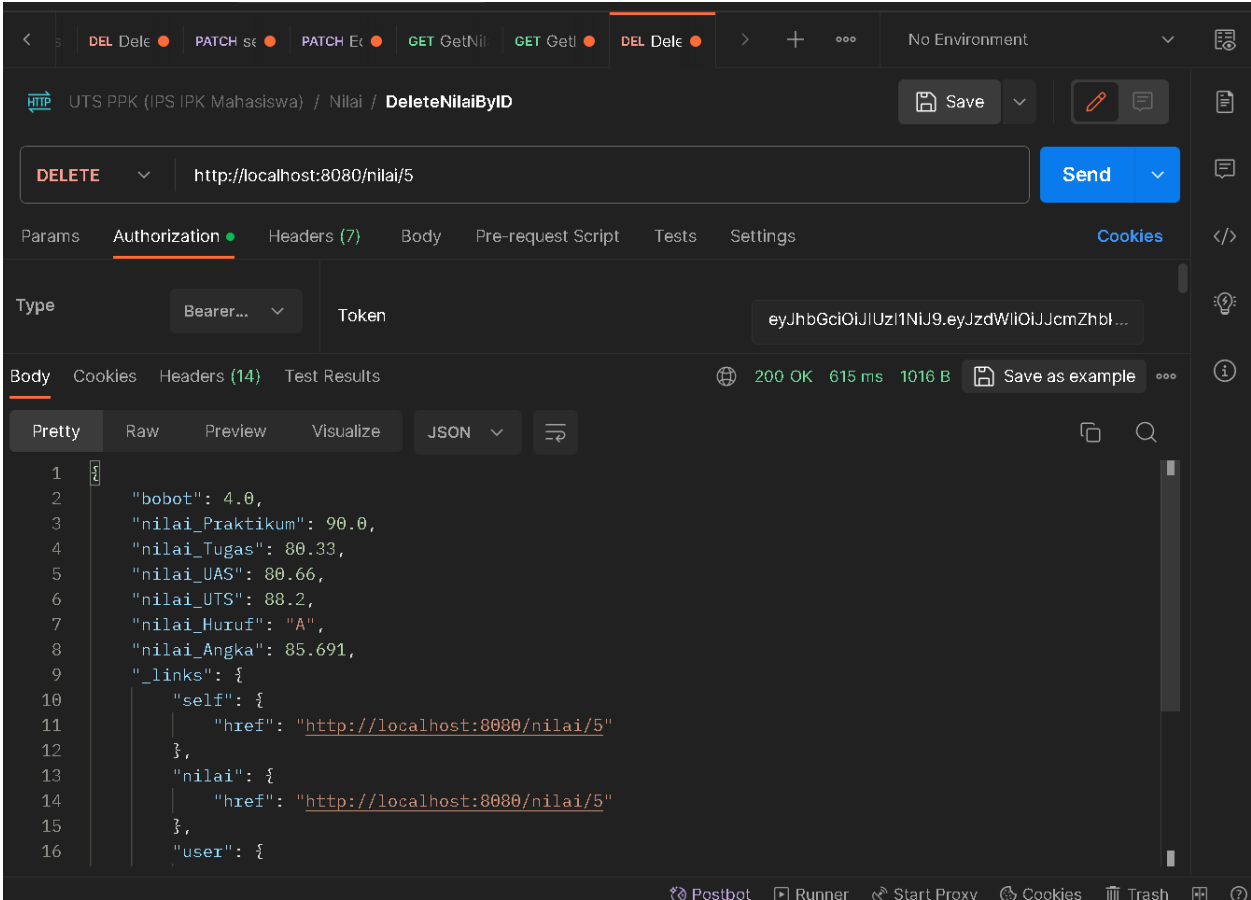
Jika yang sedang login adalah user dan id nya berbeda maka muncul pesan berikut



- *DeleteNilai*

Hanya bisa diakses oleh dosen

Pada kasus ini akan dihapus nilai dengan id 5



kelasusersmata\_kuliahuser\_matkulusersmata\_kuliahusersnilai

Limit to 1000 rows

1 • SELECT \* FROM utspk.nilai;

|   | id   | nilai_praktikum | nilai_tugas | nilai_uas | nilai_uts | matkul_id | periode_id | user_id | nilai_angka | nilai_huruf | b  |
|---|------|-----------------|-------------|-----------|-----------|-----------|------------|---------|-------------|-------------|----|
| ▶ | 1    | 90              | 93.33       | 70.66     | 80.33     | 1         | 1          | 2       | 85          | A           | 4  |
|   | 2    | 50              | 60.33       | 90.66     | 81        | 2         | 1          | 2       | 72.531      | B           | 3  |
|   | 3    | 40              | 50.33       | 60.66     | 71        | 6         | 2          | 2       | 61.18       | C           | 2  |
|   | 4    | 80              | 80.33       | 80.66     | 71        | 5         | 1          | 2       | 77.18       | B+          | 3  |
| * | NULL | NULL            | NULL        | NULL      | NULL      | NULL      | NULL       | NULL    | NULL        | NULL        | NU |

Result Grid

Filter Rows:

Edit

Export/Import

Wrap Cell Content

Result Grid

Form Editor

Field Types

Untuk edit belum sempat dibuat karena keterbatasan waktu. Namun logic simplenya perlu dibuat manual di controller karena memerlukan automatisasi nilai Huruf bobot dan nilai angka

2.5.9 IPS

- GenerateIPS

Request ini hanya bisa dilakukan oleh admin. Request ini akan menggenerate IP semester semua mahasiswa. Apabila ip dengan user dan periode tersebut sudah ada maka akan diskip proses generatennya. Jadi hanya menggenerate IP yang belum ada. Jika datanya banyak mungkin algoritma ini akan jadi problem karena konsepnya masih mengget semua data nilai dengan looping lalu dicek apakah sudah dimasukan ke Ip atau belum. Selanjutnya Jika belum baru akan dimasukkan.

UTS PPK (IPS IPK Mahasiswa) / IPS / GenerateIPS

Save

POST

http://localhost:8080/ips

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Type

Bearer...

Token

eyJhbGciOiJIUzI1NiJ9.eyJzdWwiOiJlTjdTd...

The authorization header will be automatically generated when you

Body

Cookies

Headers (14)

Test Results

200 OK

2.71 s

458 B

Save as

Pretty

Raw

Preview

Visualize

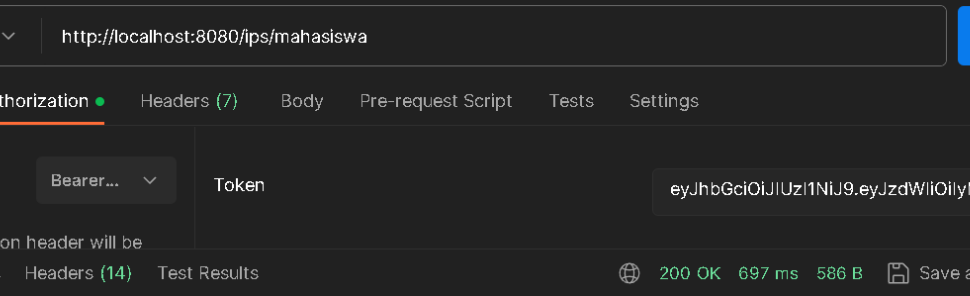
JSON

1

2

3

"message": "Berhasil generate IPS"

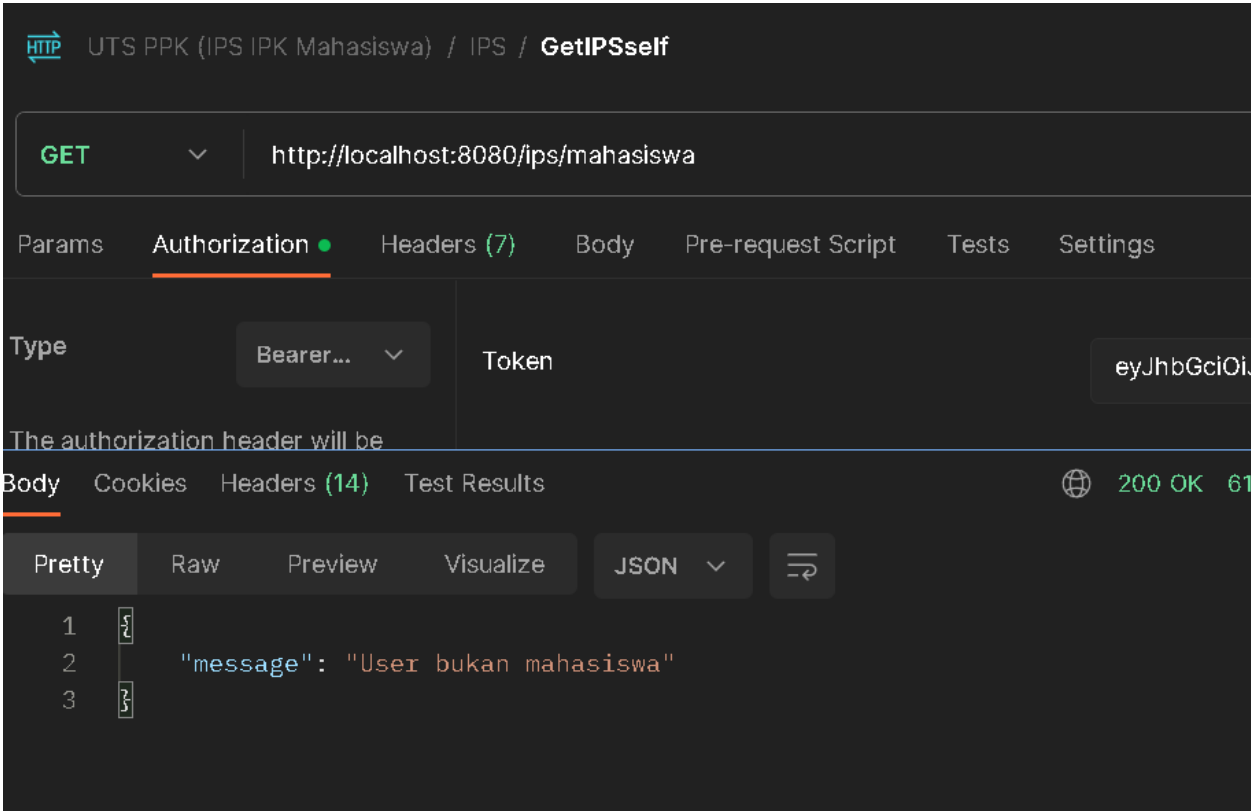


The screenshot shows a REST client interface with the following details:

- URL:** `http://localhost:8080/ips/mahasiswa`
- Method:** GET
- Authorization:** Bearer token: `eyJhbGciOiJIUzI1NiU9.eyJzdWIiOiJyMjIxMTt...`
- Response:** 200 OK, 697 ms, 586 B
- Response Body (JSON):**

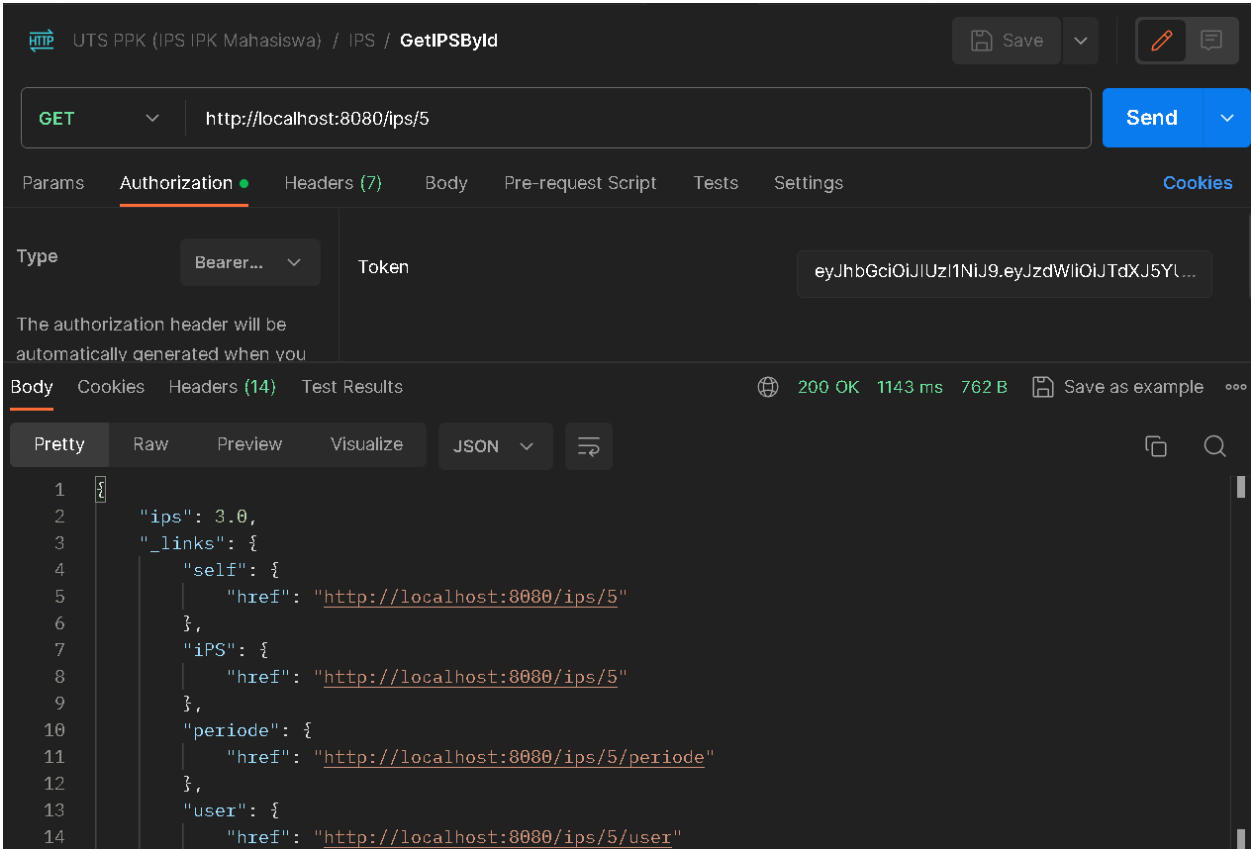
```
[{"id": 5, "ips": 3.0, "namaMahasiswa": "Aliefta Zulvansyah B", "periode": "Semester_1"}, {"id": 6, "ips": 2.0, "namaMahasiswa": "Aliefta Zulvansyah B", "periode": "Semester_2"}]
```

Jika role user active bukan mahasiswa maka akan muncul pesan seperti ini



- *GetIPsbyId*

Hanya dapat diakses dosen dan admin



- *GetIpsByMahasiswa*

Hanya dapat diakses dosen dan admin

HTTP

UTS PPK (IPS IPK Mahasiswa) / IPS / GetIPSbyMahasiswa

Save

GET

▼

http://localhost:8080/ips/mahasiswa/2

Send

▼

Params

Authorization ●

Headers (7)

Body

Pre-request Script

Tests

Settings

Cookies

Type

Bearer... ▼

Token

eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJTdXJ5YU...

The authorization header will be automatically generated when you

Body

Cookies

Headers (14)

Test Results

200 OK

875 ms

586 B

Save as example

Pretty

Raw

Preview

Visualize

JSON ▼

1

2

3

4

5

6

7

8

9

10

11

12

13

14

{

"id": 5,

"ips": 3.0,

"namaMahasiswa": "Aliefta Zulvansyah B",

"periode": "Semester\_1"

},

{

"id": 6,

"ips": 2.0,

"namaMahasiswa": "Aliefta Zulvansyah B",

"periode": "Semester\_2"

}

}

Methode delete,put patch tidak diperkenankan pada path /ips

```
// DENYALL
.requestMatchers( method: HttpMethod.DELETE, patterns: "/ips").denyAll()
.requestMatchers( method: HttpMethod.PUT, patterns: "/ips").denyAll()
.requestMatchers( method: HttpMethod.PATCH, patterns: "/ips").denyAll()
```

## **BAB III**

### **PENUTUP**

#### **3.1 Kesimpulan**

Dalam suatu lembaga pendidikan tinggi seperti Politeknik Statistika STIS, manajemen IP mahasiswa menjadi suatu hal yang sangat penting. Setiap akhir semester usai, lumrah halnya bagi mahasiswa untuk menerima indeks prestasi mereka masing masing. Maka dari itu sistem ini diharapkan dapat memfasilitasi hal tersebut

#### **3.2 Saran**

Penulis sadar masih banyak kekurangan dalam sistem ini misalnya belum ada pengecekan batas nim dan nip, belum terlalu jelas dosen pengampu matkul tertentu untuk mahasiswa tertentu, dan masih banyak lagi. Oleh karena itu, penulis terbuka terhadap segala saran dan kritik guna membuat sistem manajemen IP yang lebih baik.