
DOKUMEN TEKNIS

BACKEND

Smart English Adaptive Learning System

Daftar Isi

Daftar Isi.....	1
Daftar Gambar.....	2
Daftar Tabel.....	3
A. Overview.....	7
B. Perencanaan dan Analisis Kebutuhan.....	7
C. Use Case Diagram.....	8
D. Pemasangan dan Konfigurasi Backend.....	10
1. Prasyarat.....	10
2. Pemasangan Node.js dan NPM.....	11
3. Pemasangan Git.....	12
4. Pemasangan MySQL.....	12
5. Melakukan Clone Aplikasi Backend.....	13
6. Konfigurasi Database MySQL di Backend.....	14
7. Pemasangan dan Konfigurasi Postman.....	17
8. Pemasangan ngrok (Untuk Testing di Frontend dan Flutter).....	18
9. Testing.....	18
E. Desain Sistem Database.....	19
F. Middlewares.....	19
G. File Handling dan Media Management.....	21
H. Integration.....	24
I. Dokumentasi Endpoint Backend.....	26
1. Auth.....	26
2. User Model.....	35
3. Section Model.....	47
4. Topic Model.....	56
5. Level Model.....	72
6. Exercise Model.....	91
7. StdLearning Model.....	121
8. StdExercise Model.....	142
9. Class Model.....	152
10. Monitoring Model.....	166
11. Report Model.....	184

Daftar Gambar

Gambar 1. Website use case diagram.....	9
Gambar 2. Mobile use case diagram.....	10
Gambar 3. Create Database.....	13
Gambar 4. Setting env.....	15
Gambar 5. Migrate Tables.....	15
Gambar 6. Hasil Migrate Tables.....	16
Gambar 7. Seeder Admin User.....	16
Gambar 8. Hasil Seeder Admin User.....	16
Gambar 9. Menjalankan Server.....	17
Gambar 10. Struktur dan Relasi Database.....	19
Gambar 11. Code Authentication.....	20
Gambar 12. CORS Configuration.....	21
Gambar 13. Handling Upload Media.....	22
Gambar 14. Directory Structure (Level Model).....	23
Gambar 15. Input File Validation.....	24
Gambar 16. Nodemailer Configuration.....	25
Gambar 17. Multer Configuration.....	25

Daftar Tabel

Tabel 1. Route Register Student.....	26
Tabel 2. Parameter Register Student.....	27
Tabel 3. Status Code Register Student.....	27
Tabel 4. Route Register Teacher.....	27
Tabel 5. Parameter Register Teacher.....	28
Tabel 6. Status Code Register Teacher.....	28
Tabel 7. Route Validate Email.....	29
Tabel 8. Parameter Validate Email.....	29
Tabel 9. Status Code Validate Email.....	29
Tabel 10. Route Login.....	30
Tabel 11. Parameter Login.....	30
Tabel 12. Status Code Login.....	31
Tabel 13. Route Refresh Token.....	31
Tabel 14. Parameter Refresh Token.....	32
Tabel 15. Status Code Refresh Token.....	32
Tabel 16. Route Forgot Password.....	33
Tabel 17. Parameter Forgot Password.....	33
Tabel 18. Status Code Forgot Password.....	33
Tabel 19. Route Reset Password.....	34
Tabel 20. Parameter Reset Password.....	34
Tabel 21. Status Code Reset Password.....	35
Tabel 22. Route Get Users.....	35
Tabel 23. Status Code Get Users.....	37
Tabel 24. Route Get User by Id.....	37
Tabel 25. Parameter Get User by Id.....	37
Tabel 26. Status Code Get User by Id.....	38
Tabel 27. Route Get Me.....	38
Tabel 28. Status Code Get Me.....	39
Tabel 29. Route Get Teachers.....	39
Tabel 30. Parameter Get Teachers.....	40
Tabel 31. Status Code Get Teachers.....	41
Tabel 32. Route Get Students.....	41
Tabel 33. Parameter Get Students.....	42
Tabel 34. Status Code Get Students.....	43
Tabel 35. Route Update User.....	44
Tabel 36. Parameter Update User.....	44
Tabel 37. Status Code Update User.....	45

Tabel 38. Route Update User Password.....	45
Tabel 39. Parameter Update User Password.....	46
Tabel 40. Status Code Update User Password.....	46
Tabel 41. Route Delete User.....	46
Tabel 42. Parameter Delete User.....	47
Tabel 43. Status Code Delete User.....	47
Tabel 44. Route Get Section.....	48
Tabel 45. Status Code Get Section.....	49
Tabel 46. Route Get Section by Id.....	49
Tabel 47. Parameter Get Section by Id.....	49
Tabel 48. Status Code Get Section by Id.....	50
Tabel 49. Route Get Section For Admin Page.....	50
Tabel 50. Parameter Get Section For Admin Page.....	50
Tabel 51. Status Code Get Section For Admin Page.....	52
Tabel 52. Route Create Section.....	52
Tabel 53. Parameter Create Section.....	52
Tabel 54. Status Code Create Section.....	53
Tabel 55. Route Update Section.....	53
Tabel 56. Parameter Update Section.....	54
Tabel 57. Status Code Update Section.....	55
Tabel 58. Route Delete Section.....	55
Tabel 59. Parameter Delete Section.....	55
Tabel 60. Status Code Delete Section.....	56
Tabel 61. Route Get Topics.....	56
Tabel 62. Status Code Get Topics.....	57
Tabel 62. Route Get Topic by Id.....	58
Tabel 63. Parameter Get Topic by Id.....	58
Tabel 64. Status Code Get Topic by Id.....	59
Tabel 65. Route Get Topic by Section Id.....	59
Tabel 66. Parameter Get Topic by Section Id.....	59
Tabel 67. Status Code Get Topic by Section Id.....	60
Tabel 68. Route Get Topic For Admin Page.....	60
Tabel 69. Parameter Get Topic For Admin Page.....	61
Tabel 70. Status Code Get Topic For Admin Page.....	62
Tabel 71. Route Get Completed Topics.....	62
Tabel 72. Status Code Get Completed Topics.....	63
Tabel 73. Route Create Topic.....	63
Tabel 74. Parameter Create Topic.....	64
Tabel 75. Status Code Create Topic.....	69
Tabel 76. Route Update Topic.....	69

Tabel 77. Parameter Update Topic.....	70
Tabel 78. Status Code Update Topic.....	70
Tabel 79. Route Delete Topic.....	71
Tabel 80. Parameter Delete Topic.....	71
Tabel 81. Status Code Delete Topic.....	71
Tabel 82. Route Get Levels.....	72
Tabel 83. Status Code Get Levels.....	73
Tabel 84. Route Get Level by Id.....	73
Tabel 85. Parameter Get Level by Id.....	74
Tabel 86. Status Code Get Level by Id.....	75
Tabel 87. Route Get Level by Topic Id.....	75
Tabel 88. Parameter Get Level by Topic Id.....	75
Tabel 89. Status Code Get Level by Topic Id.....	80
Tabel 90. Route Get Level For Admin Page.....	80
Tabel 91. Parameter Get Level For Admin Page.....	81
Tabel 92. Status Code Get Level For Admin Page.....	83
Tabel 93. Route Get Level Files.....	84
Tabel 94. Parameter Get Level Files.....	84
Tabel 95. Status Code Get Level Files.....	85
Tabel 96. Route Update Level.....	85
Tabel 97. Parameter Update Level.....	86
Tabel 98. Status Code Update Level.....	87
Tabel 99. Route Upload Level File.....	87
Tabel 100. Parameter Upload Level File.....	88
Tabel 101. Status Code Upload Level File.....	89
Tabel 102. Route Delete Level File by Id and Levelname.....	89
Tabel 103. Parameter Delete Level File by Id and Levelname.....	89
Tabel 104. Status Code Delete Level File by Id and Levelname.....	90
Tabel 105. Route Delete Level.....	90
Tabel 106. Parameter Delete Level.....	90
Tabel 107. Status Code Delete Level.....	91
Tabel 108. Route Get Exercises.....	91
Tabel 109. Status Code Get Exercises.....	94
Tabel 110. Route Get Exercise by Id.....	94
Tabel 111. Parameter Get Exercise by Id.....	94
Tabel 112. Status Code Get Exercise by Id.....	95
Tabel 113. Route Get Exercise by Level Id.....	95
Tabel 114. Parameter Get Exercise by Level Id.....	96
Tabel 115. Status Code Get Exercise by Level Id.....	101
Tabel 116. Route Get Exercise by Level Id for Admin.....	101

Tabel 117. Parameter Get Exercise by Level Id for Admin.....	101
Tabel 118. Status Code Get Exercise by Level Id for Admin.....	106
Tabel 119. Route Get Exercise for Admin Page.....	106
Tabel 120. Parameter Get Exercise for Admin Page.....	107
Tabel 121. Status Code Get Exercise for Admin Page.....	109
Tabel 122. Route Create Exercise.....	109
Tabel 123. Parameter Create Exercise.....	111
Tabel 124. Status Code Create Exercise.....	113
Tabel 125. Route Update Exercise.....	114
Tabel 126. Parameter Update Exercise.....	115
Tabel 127. Status Code Update Exercise.....	117
Tabel 128. Route Update Exercise Title.....	117
Tabel 129. Parameter Update Exercise Title.....	118
Tabel 130. Status Code Update Exercise Title.....	119
Tabel 131. Route Delete Exercise File.....	119
Tabel 132. Parameter Delete Exercise File.....	120
Tabel 133. Status Code Delete Exercise File.....	120
Tabel 134. Route Delete Exercise.....	121
Tabel 135. Parameter Delete Exercise.....	121
Tabel 136. Status Code Delete Exercise.....	121
Tabel 137. Route Get StdLearnings.....	122
Tabel 138. Status Code Get StdLearnings.....	123
Tabel 139. Route Get StdLearning by Id.....	123
Tabel 140. Parameter Get StdLearning by Id.....	123
Tabel 141. Status Get StdLearning by Id.....	124
Tabel 142. Route Get Latest StdLearning Data by Level Id.....	124
Tabel 143. Parameter Get Latest StdLearning Data by Level Id.....	125
Tabel 144. Status Code Get Latest StdLearning Data by Level Id.....	125
Tabel 145. Route Get Score and Next Level by StdLearning Id.....	126
Tabel 146. Parameter Get Score and Next Level by StdLearning Id.....	126
Tabel 147. Status Code Get Score and Next Level by StdLearning Id.....	127
Tabel 148. Route Get Learning History.....	127
Tabel 149. Parameter Get Learning History.....	127
Tabel 150. Status Code Get Learning History.....	129
Tabel 151. Route Get Learning History by Section Id.....	130
Tabel 152. Parameter Get Learning History by Section Id.....	130
Tabel 153. Status Code Get Learning History by Section Id.....	132
Tabel 154. Route Get Learning History by Topic Id.....	132
Tabel 155. Parameter Get Learning History by Topic Id.....	132
Tabel 156. Status Code Get Learning History by Topic Id.....	134

Tabel 157. Route Get Recent Student Activities.....	134
Tabel 158. Parameter Get Recent Student Activities.....	134
Tabel 159. Status Code Get Recent Student Activities.....	136
Tabel 160. Route Get Recent Student Activities by Class Id.....	137
Tabel 161. Parameter Get Recent Student Activities by Class Id.....	137
Tabel 162. Status Code Get Recent Student Activities by Class Id.....	139
Tabel 163. Route Create StdLearning.....	139
Tabel 164. Parameter Create StdLearning.....	140
Tabel 165. Status Code Create StdLearning.....	140
Tabel 166. Route Student Feedback by Id StdLearning.....	141
Tabel 167. Parameter Student Feedback by Id StdLearning.....	141
Tabel 168. Status Code Student Feedback by Id StdLearning.....	142
Tabel 169. Route Get StdExercises.....	142
Tabel 170. Status Code Get StdExercises.....	145
Tabel 171. Route Get StdExercise by Id.....	145
Tabel 172. Parameter Get StdExercise by Id.....	145
Tabel 173. Status Code Get StdExercise by Id.....	146
Tabel 174. Route Get Student Answer by StdLearning Id.....	146
Tabel 175. Parameter Get Student Answer by StdLearning Id.....	146
Tabel 176. Status Code Get Student Answer by StdLearning Id.....	150
Tabel 177. Route Get Student Answers.....	150
Tabel 178. Parameter Get Student Answers.....	151
Tabel 179. Status Code Get Student Answers.....	152
Tabel 180. Route Get Classes.....	152
Tabel 181. Status Code Get Classes.....	153
Tabel 182. Route Get Class by Id.....	154
Tabel 183. Parameter Get Class by Id.....	154
Tabel 184. Status Code Get Class by Id.....	154
Tabel 185. Route Get Class for Admin.....	155
Tabel 186. Parameter Get Class for Admin.....	155
Tabel 187. Status Code Get Class for Admin.....	156
Tabel 188. Route Get Student by Class Id.....	157
Tabel 189. Parameter Get Student by Class Id.....	157
Tabel 190. Status Code Get Student by Class Id.....	158
Tabel 191. Route Get Student With No Class.....	158
Tabel 192. Status Code Get Student With No Class.....	160
Tabel 193. Route Create Class.....	161
Tabel 194. Parameter Create Class.....	161
Tabel 195. Status Code Create Class.....	162
Tabel 196. Route Update Student Class by Name and NISN.....	162

Tabel 197. Parameter Update Student Class by Name and NISN.....	162
Tabel 198. Status Code Update Student Class by Name and NISN.....	164
Tabel 199. Route Update Class.....	164
Tabel 200. Parameter Update Class.....	164
Tabel 201. Status Code Update Class.....	165
Tabel 202. Route Delete Class.....	165
Tabel 203. Parameter Delete Class.....	166
Tabel 204. Status Code Delete Class.....	166
Tabel 205. Route Get Monitorings.....	167
Tabel 206. Status Code Get Monitorings.....	168
Tabel 207. Route Get Monitoring by Id.....	168
Tabel 208. Parameter Get Monitoring by Id.....	168
Tabel 209. Status Code Get Monitoring by Id.....	169
Tabel 210. Route Monitoring Student Progress.....	169
Tabel 211. Parameter Monitoring Student Progress.....	170
Tabel 212. Status Code Monitoring Student Progress.....	171
Tabel 213. Route Monitoring Student Progress by Id.....	172
Tabel 214. Parameter Monitoring Student Progress by Id.....	172
Tabel 215. Status Code Monitoring Student Progress by Id.....	173
Tabel 216. Route Download Monitoring Student Progress to CSV by Monitoring Id.....	174
Tabel 217. Parameter Download Monitoring Student Progress to CSV by Monitoring Id....	174
Tabel 218. Status Code Download Monitoring Student Progress to CSV by Monitoring Id.	174
Tabel 219. Route Monitoring Feedback.....	175
Tabel 220. Parameter Monitoring Feedback.....	175
Tabel 221. Status Code Monitoring Feedback.....	176
Tabel 222. Route Get Monitoring Data by Topic Id.....	176
Tabel 223. Parameter Get Monitoring Data by Topic Id.....	176
Tabel 224. Status Code Get Monitoring Data by Topic Id.....	177
Tabel 225. Route Get Class Monitoring by Class Id.....	177
Tabel 226. Parameter Get Class Monitoring by Class Id.....	178
Tabel 227. Status Code Get Class Monitoring by Class Id.....	179
Tabel 228. Route Get Class Monitoring by Class & Topic Id.....	179
Tabel 229. Parameter Get Class Monitoring by Class & Topic Id.....	180
Tabel 230. Status Code Get Class Monitoring by Class & Topic Id.....	181
Tabel 231. Route Get Monitoring Feedback by Class.....	182
Tabel 232. Parameter Get Monitoring Feedback by Class.....	182
Tabel 233. Status Code Get Monitoring Feedback by Class.....	183
Tabel 234. Route Download Class Monitoring Data to CSV by Class and Topic Id.....	183
Tabel 235. Parameter Download Class Monitoring Data to CSV by Class and Topic Id.....	184
Tabel 236. Status Code Download Class Monitoring Data to CSV by Class and Topic Id...	184

Tabel 237. Route Get Reports.....	184
Tabel 238. Parameter Get Reports.....	185
Tabel 239. Status Code Get Reports.....	186
Tabel 240. Route Get Five New Reports.....	186
Tabel 241. Status Code Get Five New Reports.....	188
Tabel 242. Route Get Report By Id.....	188
Tabel 243. Parameter Get Report By Id.....	188
Tabel 244. Status Code Get Report By Id.....	189
Tabel 245. Route Get Report By User Id.....	189
Tabel 246. Parameter Get Report By User Id.....	190
Tabel 247. Status Code Get Report By User Id.....	190
Tabel 248. Route Create Report.....	191
Tabel 249. Parameter Create Report.....	191
Tabel 250. Status Code Create Report.....	192

A. Overview

SEALS adalah sebuah aplikasi pembelajaran bahasa Inggris adaptif yang dirancang khusus untuk siswa SMK Pariwisata dan Bisnis. Aplikasi ini dikembangkan untuk menyediakan pengalaman belajar yang disesuaikan dengan kebutuhan individu siswa, memanfaatkan teknologi *adaptive learning*. Dengan SEALS, siswa dapat mengakses konten belajar yang relevan dengan industri pariwisata dan bisnis, melalui perangkat *web* maupun *mobile*.

Aplikasi SEALS memungkinkan siswa untuk belajar secara mandiri dengan memilih *section* dan *topic* yang sesuai dengan minat dan tingkat kemampuan mereka. Berdasarkan hasil *pretest*, sistem akan menempatkan siswa pada level yang tepat, dan secara otomatis menyesuaikan materi pembelajaran yang sesuai dengan hasil tersebut. SEALS juga menyediakan berbagai fitur interaktif seperti latihan soal dan ujian untuk mengukur kemajuan belajar siswa.

Aplikasi SEALS memiliki fungsi manajemen bagi *admin* dan guru untuk mengelola data siswa, materi pembelajaran, serta memantau kemajuan belajar siswa melalui laporan dalam bentuk tabel. *Admin* memiliki kontrol penuh atas sistem, termasuk kemampuan untuk mengelola data guru, siswa, *section*, *topic*, dan pengaturan *rule adaptive learning*.

SEALS diharapkan dapat meningkatkan keterlibatan siswa dalam belajar bahasa Inggris dan membantu mereka mencapai kompetensi yang sesuai dengan kebutuhan industri pariwisata dan bisnis, dengan memberikan fleksibilitas dalam metode pembelajaran yang dapat diakses di mana pun dan kapan pun.

B. Perencanaan dan Analisis Kebutuhan

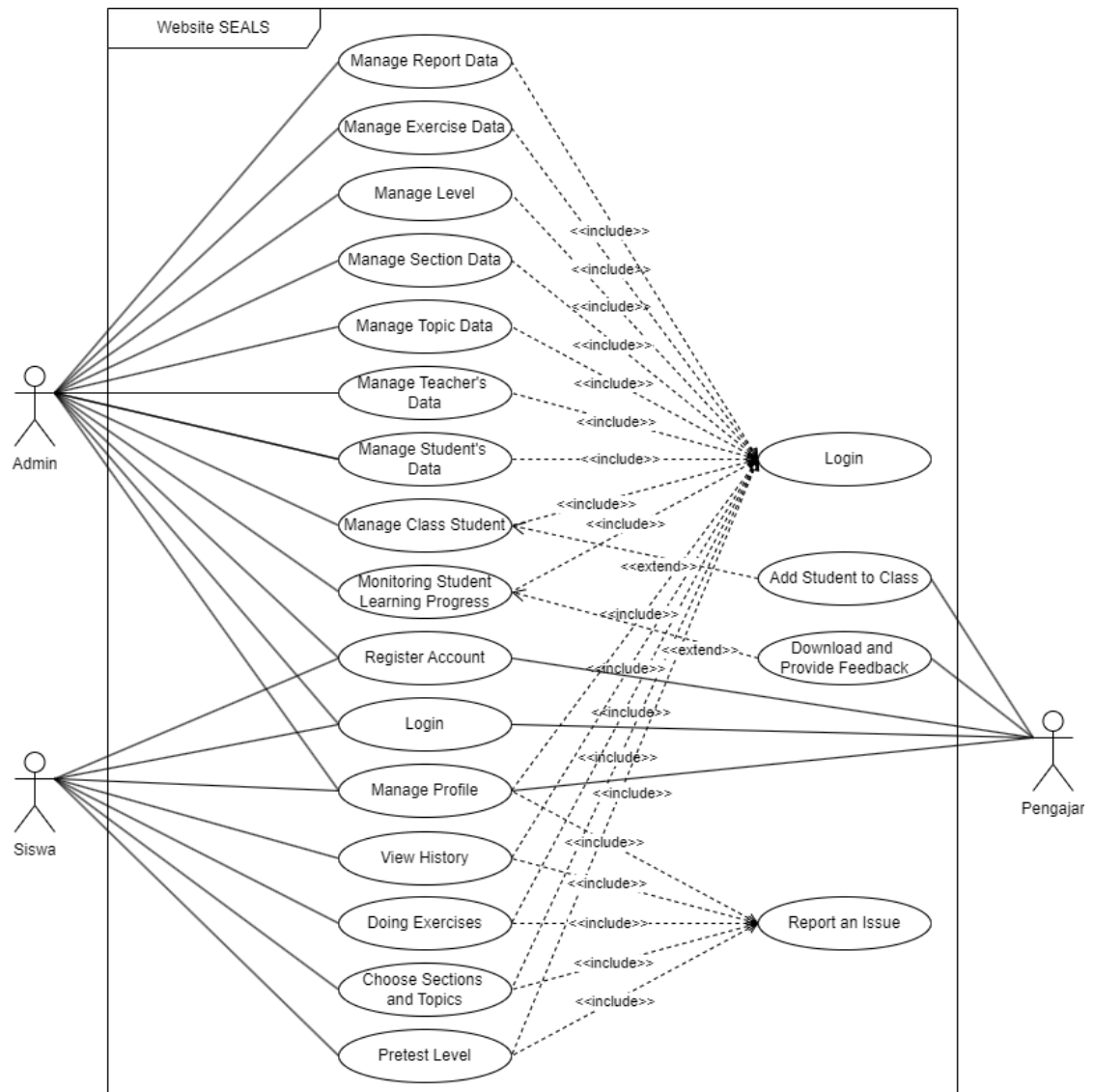
Dalam tahap desain dibutuhkan antarmuka berbasis *web* dan *mobile* dengan fokus pada pengalaman pengguna yang optimal. Desain UI dirancang agar menarik secara visual dan mudah digunakan, menyesuaikan dengan karakteristik siswa SMK, serta memilih skema warna yang sesuai dengan jurusan industri pariwisata dan bisnis. Tipografi dipilih agar mudah dibaca, dengan penambahan elemen visual seperti gambar dan ilustrasi yang relevan. Desain ini dioptimalkan untuk tampil dengan baik

di layar *web* dan *mobile*, serta diuji untuk memastikan kemudahan penggunaannya oleh siswa SMK.

Sementara itu, dalam desain UX, alur interaksi pengguna diatur agar navigasi aplikasi menjadi lebih mudah dan memberikan panduan yang jelas. Fitur tambahan seperti personalisasi, gamifikasi, dan elemen interaktif dimasukkan untuk mendukung pembelajaran pengguna dengan lebih efektif. Hasil dari proses ini meliputi *prototype* UI/UX aplikasi, panduan visual, dan dokumentasi alur interaksi pengguna.

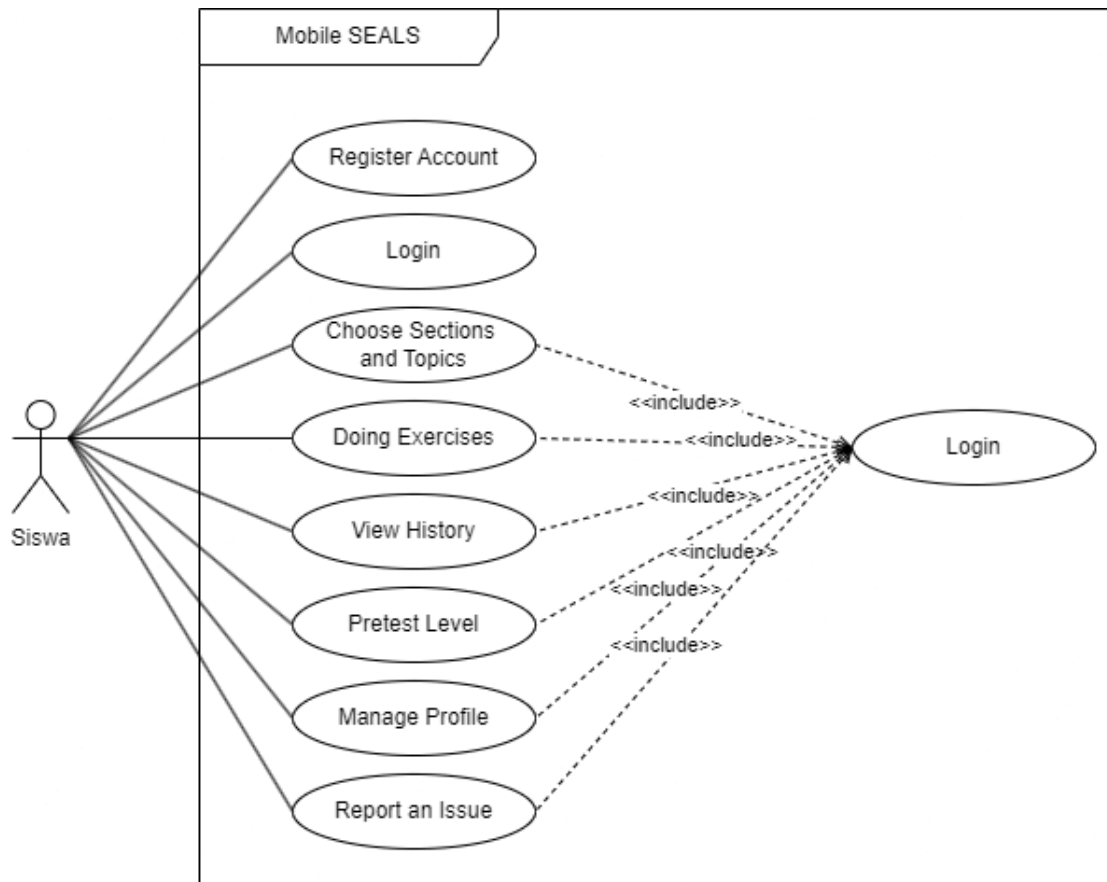
C. Use Case Diagram

Website SEALS mendukung tiga *role user* yang di antaranya adalah *admin*, pengajar, dan siswa sesuai yang sudah digambarkan pada diagram di bawah ini. *Admin* dapat mengelola berbagai data, mulai dari *user* yaitu pengajar dan siswa, serta menambahkan bahan pembelajaran dan latihan. Pengajar bisa menambah siswa ke kelas, memantau progres belajar siswa, dan memberikan umpan balik. Sedangkan siswa dapat belajar sesuai yang diinginkan, mengerjakan soal, melihat *history* pengerjaan latihan, dan juga melaporkan masalah.



Gambar 1. *Website use case diagram*

Sementara itu, pada *mobile* hanya *role* siswa yang akan bisa menggunakan aplikasi. Jadi, aktivitasnya kurang lebih sama seperti yang telah tercantum di *use case diagram* bagian *website* untuk *role* siswa.



Gambar 2. Mobile use case diagram

D. Pemasangan dan Konfigurasi *Backend*

Dokumentasi ini bertujuan untuk memberikan panduan tentang pemasangan serta konfigurasi sebelum dapat menggunakan aplikasi backend untuk SEALS ini. Dibawah adalah beberapa poin penting sebelum dapat menggunakan aplikasi *backend* ini:

1. Prasyarat

Sebelum memulai pemasangan backend pada perangkat, pastikan telah menyiapkan alat dan perangkat lunak berikut untuk mendukung pemasangan dan konfigurasi backend:

- Sistem Operasi: Windows, Linux.
- Node.js: Platform runtime JavaScript yang diperlukan untuk menjalankan server backend. Versi yang direkomendasikan adalah 20.x atau lebih baru.
- NPM (Node Package Manager): Manajer paket yang disertakan dalam instalasi Node.js, paket ini digunakan untuk mengelola dependensi.

- d. MySQL: Sistem manajemen basis data relasional (RDBMS) untuk penyimpanan data ke database.
- e. Git: Alat kontrol yang digunakan untuk mengelola versi kode serta kolaborasi dengan tim.
- f. Postman: Alat pengujian API yang digunakan untuk memvalidasi endpoint pada backend.
- g. Visual Studio Code: Editor kode yang disarankan, mendukung pengembangan berbasis Node.js dan integrasi dengan Git

2. Pemasangan Node.js dan NPM

Node.js diperlukan untuk menjalankan server backend berbasis Express.js yang digunakan pada sistem SEALS ini. Berikut adalah langkah-langkah pemasangan node.js yang bisa diikuti:

- a. Download Node.js
 - Kunjungi [official website node.js](https://nodejs.org/) dan unduh versi LTS terbaru (Long Term Support) dari website tersebut.
 - Versi LTS ini lebih direkomendasikan karena stabilitasnya.
- b. Instalasi Node.js
 - Jalankan installer yang telah diunduh lalu ikuti panduan instalasi dari installer tersebut.
 - Pastikan opsi untuk menambahkan Node.js ke PATH dicentang, agar perintah Node.js dapat digunakan di terminal.
- c. Verifikasi Instalasi
 - Buka terminal atau command prompt, lalu jalankan perintah `npm -v` dan `node -v` untuk memastikan bahwa Node.js dan NPM sudah terpasang dengan benar pada perangkat.
 - Jika instalasi berhasil, kedua perintah di atas akan menampilkan versi Node.js dan NPM yang telah terpasang pada perangkat.

3. Pemasangan Git

Git diperlukan untuk melakukan manajemen versi kode sumber, git disini juga digunakan untuk melakukan clone kode backend yang telah dipush ke gitlab. Berikut langkah-langkah pemasangan git ke perangkat:

- a. Download Git
 - Kunjungi [Git official website](#) dan unduh installer untuk sistem operasi yang digunakan pada perangkat.
- b. Instalasi Git
 - Jalankan installer Git dan ikuti panduan instalasinya.
 - Pada bagian "Adjusting your PATH environment", pastikan opsi "Git from the command line" dipilih.
- c. Verifikasi Instalasi
 - Buka terminal atau command prompt dan ketik perintah git --version untuk memverifikasi instalasi apakah sudah terinstall pada perangkat atau belum.
 - Perintah tersebut akan menampilkan versi Git yang terpasang jika Git telah terpasang pada perangkat.
- d. Konfigurasi Awal Git
 - Setelah Git terpasang, lakukan konfigurasi awal dengan menambahkan nama pengguna dan email pada terminal.

4. Pemasangan MySQL

MySQL adalah sistem manajemen basis data relasional yang akan digunakan untuk menyimpan data dari aplikasi SEALS ini. Berikut adalah tata cara instalasi aplikasi MySQL:

- a. Download MySQL
 - Hal pertama yang perlu dilakukan adalah melakukan download aplikasi MySQL. Untuk pilihan aplikasi, bisa menggunakan XAMPP, Laragon, atau MySQL Workbench.
- b. Instalasi MySQL
 - Ikuti instruksi instalasi yang sesuai dengan sistem operasi yang digunakan pada perangkat.

- Saat instalasi, akan diminta untuk mengatur password root. Password root ini bisa dikosongkan atau diisi dengan password yang diinginkan. Simpan password ini karena akan digunakan untuk mengakses database nantinya.

c. Konfigurasi Database

- Buat database baru untuk sistem SEALS ini dengan menginputkan (CREATE DATABASE nama_database;) pada sql.
- Ganti (nama_database) dengan nama yang relevan dengan sistem SEALS ini (disarankan adaptive_learning).

```
PS C:\Elang 7\english-adaptive-learning\backend> npm run db:create

> backend@1.0.0 db:create
> npx sequelize-cli db:create --config config/config.js

Sequelize CLI [Node: 20.11.1, CLI: 6.6.2, ORM: 6.37.3]

Loaded configuration file "config\config.js".
Using environment "development".
Database adaptive_testing created.
PS C:\Elang 7\english-adaptive-learning\backend> |
```

Gambar 3. *Create Database*

5. Melakukan Clone Aplikasi Backend

Setelah semua aplikasi tadi telah terpasang, langkah berikutnya adalah melakukan clone pada aplikasi backend. Berikut adalah cara melakukan clone aplikasi backendnya:

a. Clone Repositori Git

- Clone kode proyek dari repositori Git dibawah ini lalu inputkan perintah tersebut pada terminal vscode (jangan lupa untuk menentukan direktorinya terlebih dahulu).

```
(git clone http://192.168.60.79:3000/elangptra/backend_adaptive_learning.git)
```

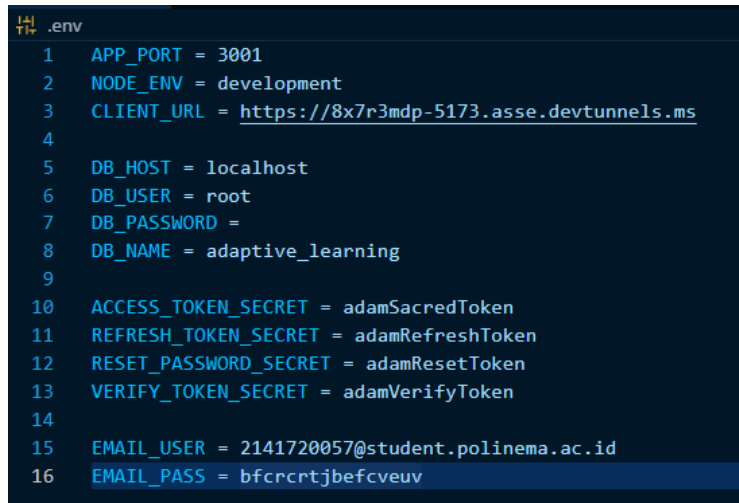
b. Masuk ke Direktori Proyek

- Pilih folder lalu buka direktori backend yang sudah di clone sebelumnya dengan menggunakan code editor (disarankan vscode).
- c. Install Dependensi
 - Install semua dependensi yang diperlukan oleh proyek Node.js dengan menjalankan perintah dibawah ini pada terminal direktori yang telah di clone tadi.
(npm install)

6. Konfigurasi Database MySQL di Backend

Setelah MySQL diinstal, database dibuat, dan sudah melakukan clone, aplikasi backend ini masih perlu dikonfigurasi agar terhubung dengan database.

- a. Buat file .env
 - Jika sudah melakukan clone aplikasi backend, bisa langsung mengubah nama file .envexample menjadi .env saja dan lengkapi isi data .env yang diperlukan.
- b. Konfigurasi Koneksi Database
 - Isi baris-baris dibawah ini pada file .env yang telah diubah tadi dengan konfigurasi MySQL yang telah diinstal sebelumnya.
(DB_HOST, DB_USER, DB_PASSWORD, dan DB_NAME).
 - Berikan isi DB_NAME dengan nama database yang telah dibuat sebelumnya dan DB_PASSWORD dengan password MySQL yang sebelumnya telah dibuat (kosongkan jika pada proses instalasi, MySQL tidak diberikan password).



```

1 APP_PORT = 3001
2 NODE_ENV = development
3 CLIENT_URL = https://8x7r3mdp-5173.asse.devtunnels.ms
4
5 DB_HOST = localhost
6 DB_USER = root
7 DB_PASSWORD =
8 DB_NAME = adaptive_learning
9
10 ACCESS_TOKEN_SECRET = adamSacredToken
11 REFRESH_TOKEN_SECRET = adamRefreshToken
12 RESET_PASSWORD_SECRET = adamResetToken
13 VERIFY_TOKEN_SECRET = adamVerifyToken
14
15 EMAIL_USER = 2141720057@student.polinema.ac.id
16 EMAIL_PASS = bfcrctjbefcveuv

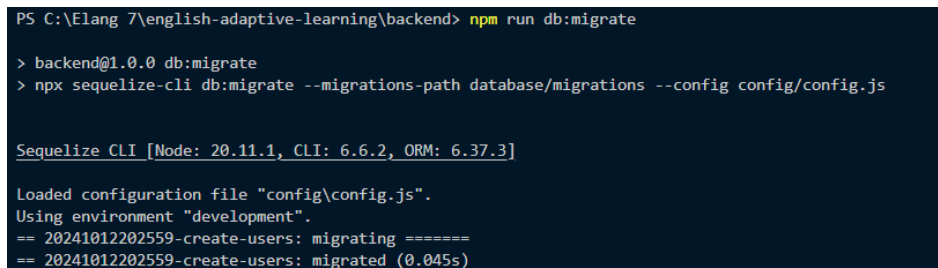
```

Gambar 4. *Setting env*

c. Migrasi Database

- Migrasi disini digunakan untuk membuat tabel-tabel yang diperlukan untuk sistem backend ini pada database yang telah dibuat dan diinputkan pada .env sebelumnya. Bisa dengan mengetikkan fungsi dibawah pada terminal vscode.

(npm run db:migrate)



```

PS C:\Elang 7\english-adaptive-learning\backend> npm run db:migrate

> backend@1.0.0 db:migrate
> npx sequelize-cli db:migrate --migrations-path database/migrations --config config/config.js

Sequelize CLI [Node: 20.11.1, CLI: 6.6.2, ORM: 6.37.3]

Loaded configuration file "config\config.js".
Using environment "development".
== 20241012202559-create-users: migrating =====
== 20241012202559-create-users: migrated (0.045s)

```

Gambar 5. *Migrate Tables*

Table	Action	Rows	Type	Collation	Size	Overhead
☐ admin_exercise	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KIB	-
☐ class	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KIB	-
☐ level	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	48.0 KIB	-
☐ matching_pairs	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KIB	-
☐ monitoring	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	64.0 KIB	-
☐ multiple_choices	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KIB	-
☐ report	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KIB	-
☐ section	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KIB	-
☐ sequelizemeta	★ Browse Structure Search Insert Empty Drop	15	InnoDB	utf8_unicode_ci	32.0 KIB	-
☐ student	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	64.0 KIB	-
☐ student_exercise	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	48.0 KIB	-
☐ student_learning	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	64.0 KIB	-
☐ teacher	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	48.0 KIB	-
☐ topic	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KIB	-
☐ true_false	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KIB	-
☐ users	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KIB	-
16 tables	Sum	15	InnoDB	utf8mb4_general_ci	624.0 KIB	0 B

Gambar 6. Hasil *Migrate Tables*

d. Melakukan Seed Data User Admin

- Seed data user admin akan membuat sebuah user dengan role admin yang digunakan untuk mengakses halaman admin. Bisa dengan mengetikkan fungsi dibawah ini pada terminal vscode.
(npm run db:seed)

```
PS C:\Elang 7\english-adaptive-learning\backend> npm run db:seed

> backend@1.0.0 db:seed
> npx sequelize-cli db:seed:all --seeders-path database/seeder --config config/config.js

Sequelize CLI [Node: 20.11.1, CLI: 6.6.2, ORM: 6.37.3]

Loaded configuration file "config\config.js".
Using environment "development".
== 20241012193900-create-admin-user: migrating =====
== 20241012193900-create-admin-user: migrated (0.263s)
```

Gambar 7. *Seeder Admin User*

ID	NAME_USERS	EMAIL	PASSWORD	ROLE	PICTURE	REFRESH_TOKEN	TIME_USERS
bc398e3c-a6f4-4c77-9ee4-a2e089ed530	Administrator	adminseals@gmail.com	\$2b\$10\$leRbagKOrTHJT3YVvBYOWVthCXsBGpCbV4RscQd0...	admin	NULL	NULL	2024-10-15 01:16:01

Gambar 8. Hasil *Seeder Admin User*

e. Menjalankan Server

- Untuk menjalankan server backend, bisa menggunakan perintah dibawah ini pada terminal direktori aplikasi backend ini.
(npm run api-start)

```
PS C:\Elang 7\english-adaptive-learning\backend> npm run api-start  
  
> backend@1.0.0 api-start  
> nodemon index.js  
  
[nodemon] 3.1.4  
[nodemon] to restart at any time, enter `rs`  
[nodemon] watching path(s): *.*  
[nodemon] watching extensions: js,mjs,cjs,json  
[nodemon] starting `node index.js`  
Server running on port http://localhost:3001  
Database connected
```

Gambar 9. Menjalankan Server

7. Pemasangan dan Konfigurasi Postman

Postman merupakan sebuah aplikasi yang digunakan untuk melakukan testing endpoint API tanpa perlu membuat tampilannya terlebih dahulu, sangat cocok untuk melakukan test API backend. Berikut adalah cara menginstall Postman pada perangkat:

a. Unduh Postman

- Kunjungi [website resmi postman](#) untuk mengunduh installer postman, sesuaikan dengan sistem operasi perangkat.
- Setelah mengunduh installer, ikuti panduan instalasi postman tersebut hingga selesai.

b. Membuat Workspace pada Postman

- Setelah Postman terpasang, buka aplikasi dan buatlah workspace baru dan berikan nama sesuai dengan sistem backend yang akan dilakukan testing (berikan nama SEALS).
- Tambahkan koleksi API yang ada pada route backend ke Postman untuk dilakukan testing.

c. Penggunaan Environment Postman

- Gunakan fitur environment di Postman untuk menyimpan variabel seperti URL backend (<http://localhost:3000>) dan token autentikasi yang dapat digunakan dalam pengujian API.

8. Pemasangan ngrok (Untuk Testing di Frontend dan Flutter)

Ngrok disini digunakan untuk keperluan testing endpoint backend ke frontend serta aplikasi flutter yang dikembangkan bersamaan dengan aplikasi backend ini. Ngrok akan melakukan forward port dan mengembalikan dalam bentuk link yang bisa diakses oleh perangkat lain selama device penyedia server tetap aktif. Dibawah ini adalah cara pemasangannya:

a. Install ngrok

- Install ngrok secara global pada terminal. Dibawah ini merupakan perintah yang bisa dijalankan untuk melakukan instalasi ngrok secara global.
(`npm install -g ngrok`)

b. Jalankan ngrok

- Jalankan ngrok untuk melakukan forward port backend (misalnya port 3001), jalankan perintah dibawah ini pada terminal direktori aplikasi backendnya.
(`ngrok http 3001`)

c. Gunakan URL ngrok

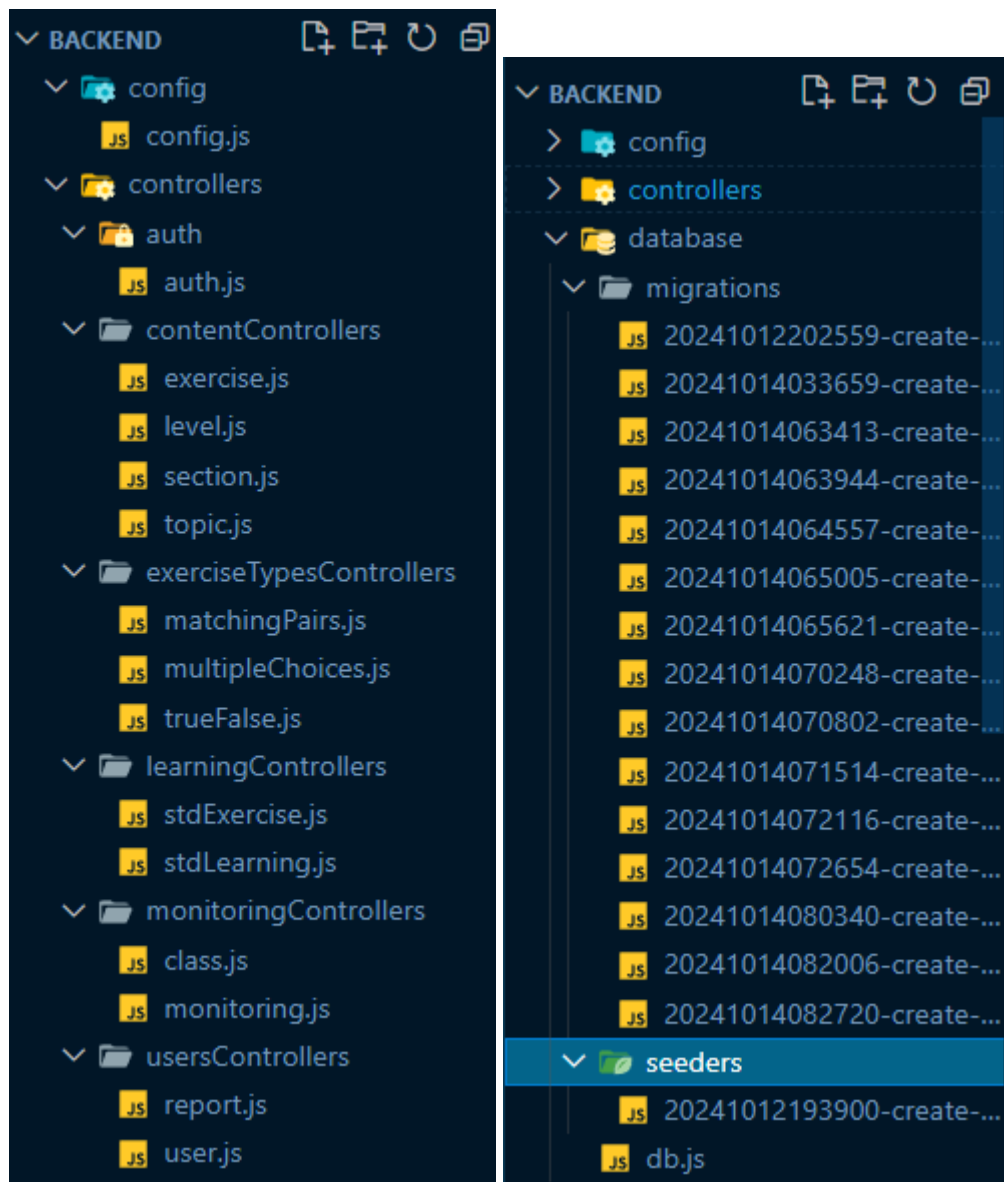
- ngrok akan memberikan URL publik (misalnya <http://abcd1234.ngrok.io>) yang dapat digunakan untuk mengakses backend dari device lain.

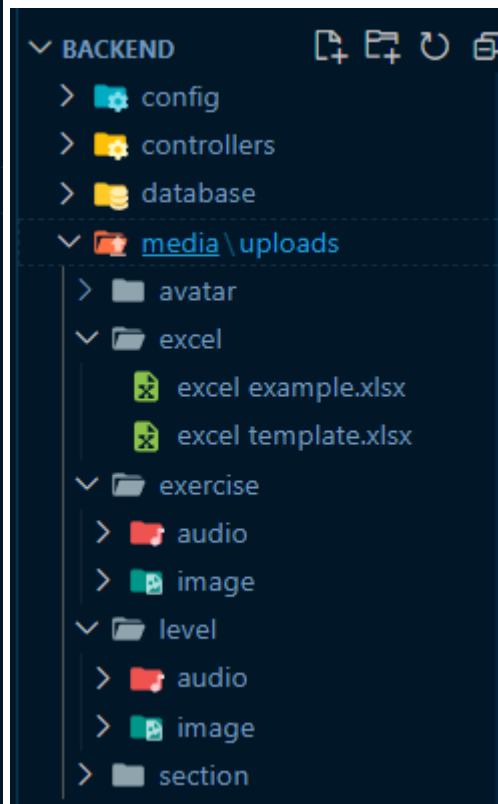
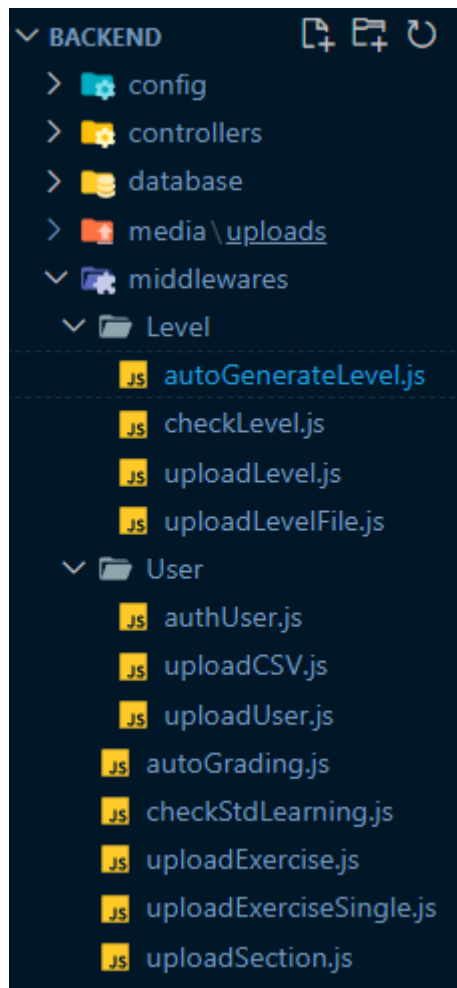
9. Testing

Untuk melakukan testing endpoint backend ini, terdapat beberapa pendekatan yang bisa digunakan. Pendekatan tersebut yaitu:

- a. Menggunakan Postman untuk melakukan testing endpoint API.
- b. atau, Menggunakan ngrok atau port forwarding lainnya agar bisa melakukan testing endpoint pada aplikasi frontend ataupun flutter.

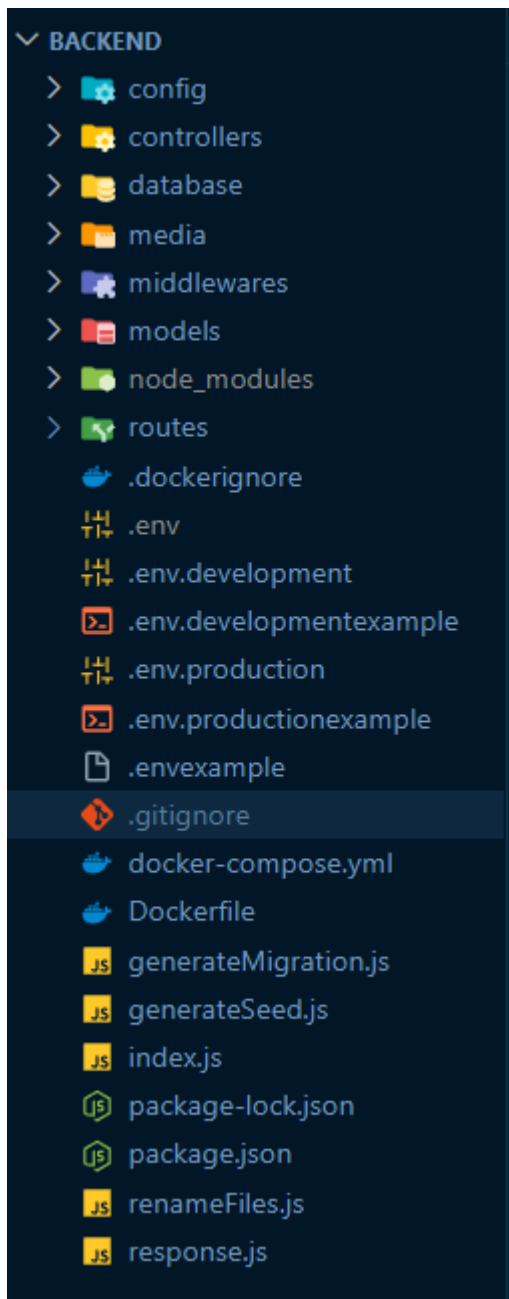
E. Struktur Proyek





- ✓ BACKEND
 - >  config
 - >  controllers
 - >  database
 - >  media
 - >  middlewares
 - ✓  models
 - ✓  contentModels
 -  exerciseModel.js
 -  levelModel.js
 -  sectionModel.js
 -  topicModel.js
 - ✓  exerciseTypesModels
 -  matchingPairsModel.js
 -  multipleChoicesModel.js
 -  trueFalseModel.js
 - ✓  learningModels
 -  stdExerciseModel.js
 -  stdLearningModel.js
 - ✓  monitoringModels
 -  classModel.js
 -  monitoringModel.js
 - ✓  usersModels
 -  reportModel.js
 -  studentModel.js
 -  teacherModel.js
 -  userModel.js
 -  index.js
 - >  node_modules

- ✓ BACKEND
 - >  config
 - >  controllers
 - >  database
 - >  media
 - >  middlewares
 - >  models
 - >  node_modules
 - ✓  routes
 - ✓  auth
 -  auth.js
 - ✓  contents
 -  exercise.js
 -  level.js
 -  section.js
 -  topic.js
 - ✓  learning
 -  stdExercise.js
 -  stdLearning.js
 - ✓  monitoring
 -  class.js
 -  monitoring.js
 - >  user
 -  index.js



F. Desain Sistem *Database*

Penulis merancang skema *database* untuk memastikan data tersimpan secara efisien dan terstruktur.


```

export const verifyLoginUser = async (req, res, next) => {
  const authHeader = req.headers.authorization;

  if (!authHeader || !authHeader.startsWith("Bearer ")) {
    return response(401, null, "Please log in to your account first!", res);
  }

  const token = authHeader.split(" ")[1];

  try {
    const decoded = jwt.verify(token, process.env.ACCESS_TOKEN_SECRET);

    const user = await models.User.findByPk(decoded.ID, {
      attributes: {
        exclude: ["PASSWORD"],
      },
    });

    if (!user) {
      return response(404, null, "User not found!", res);
    }

    req.user = user;

    next();
  } catch (error) {
    if (error.name === "TokenExpiredError") {
      return response(401, null, "Session expired. Please log in again.", res);
    } else if (error.name === "JsonWebTokenError") {
      return response(403, null, "Invalid token!", res);
    } else {
      return response(500, null, "An error occurred on the server!", res);
    }
  }
};

```

Gambar 11. *Code Authentication*

- b. *CORS Configuration*: Mengatur *Cross-Origin Resource Sharing* untuk mendukung komunikasi antara *frontend* dan *backend*.



```
const corsOptions = {
  origin: `${process.env.CLIENT_URL}`,
  methods: ["GET", "POST", "PUT", "DELETE", "OPTIONS"],
  allowedHeaders: ["Content-Type", "Authorization"],
  credentials: true,
};

app.use(cors(corsOptions));

app.use(cookieParser());
app.use(express.json());
app.use(express.urlencoded({ extended: true }));
app.use(router);
app.use("/api", express.static("media"));
```

Gambar 12. *CORS Configuration*

- c. *Error Handling: Endpoint* menangkap dan menangani kesalahan secara konsisten, sehingga mempermudah *debugging*.

H. *File Handling dan Media Management*

Penulis mengimplementasikan fitur untuk menangani unggahan dan pengelolaan *file*, termasuk:

- a. *File Upload*: Menggunakan Multer untuk mengelola unggahan *file* seperti dokumen, gambar dan *audio*.

```

const handleUpload = (req, res, next) => {
  upload(req, res, (err) => {
    if (err) {
      return response(400, null, err.message, res);
    }

    const files = req.files;
    const AUDIO = files?.AUDIO ? files.AUDIO[0] : null;
    const IMAGE = files?.IMAGE ? files.IMAGE[0] : null;

    try {
      let validFiles = true;
      let errorMessages = [];

      if (AUDIO && AUDIO.size > 10 * 1024 * 1024) {
        validFiles = false;
        AUDIO.buffer = null;
        errorMessages.push("Audio file exceeds the size limit of 10MB");
      }

      if (IMAGE && IMAGE.size > 5 * 1024 * 1024) {
        validFiles = false;
        IMAGE.buffer = null;
        errorMessages.push("Image file exceeds the size limit of 5MB");
      }

      if (validFiles) {
        req.filesToSave = { AUDIO, IMAGE };
        next();
      } else {
        clearFileBuffers({ AUDIO, IMAGE });
        return response(400, null, errorMessages.join("; "), res);
      }
    } catch (error) {
      console.log(error);
      clearFileBuffers({ AUDIO, IMAGE });
      return response(500, null, "Internal Server Error", res);
    }
  });
};

```

Gambar 13. *Handling Upload Media*

- b. *Media Storage*: Mengatur penyimpanan *file* di *server* dengan struktur direktori yang terorganisir.

```

export const saveFileToDisk = (file, type, topicId, sectionId, levelId) => {
  const ext = path.extname(file.originalname);
  const hash = generateHash(sectionId, topicId, file.originalname, file.buffer.length);
  const filename = `${type}-${levelId}-${hash}${ext}`;

  let folderPath;
  switch (type) {
    case "AUDIO":
      folderPath = path.join(process.cwd(), "media/uploads/level/audio");
      break;
    case "IMAGE":
      folderPath = path.join(process.cwd(), "media/uploads/level/image");
      break;
    default:
      folderPath = path.join(process.cwd(), "media/uploads/level");
  }

  if (!fs.existsSync(folderPath)) {
    fs.mkdirSync(folderPath, { recursive: true });
  }

  const filepath = path.join(folderPath, filename);
  fs.writeFileSync(filepath, file.buffer);
  return filename;
};

```

Gambar 14. *Directory Structure (Level Model)*

- c. *Validation*: Menambahkan validasi *file* (jenis dan ukuran) untuk mencegah unggahan yang tidak *valid*.

```

const fileFilter = (req, file, cb) => {
  const ext = path.extname(file.originalname).toLowerCase();

  switch (file.fieldname) {
    case "AUDIO":
      if (ext === ".mp3") {
        cb(null, true);
      } else {
        cb(
          new Error(
            "Invalid file type, only .mp3 files are allowed for audio!"
          ),
          false
        );
      }
      break;

    case "IMAGE":
      if (ext === ".jpg" || ext === ".jpeg" || ext === ".png") {
        cb(null, true);
      } else {
        cb(
          new Error(
            "Invalid file type, only .jpg, .jpeg, and .png files are allowed for image!"
          ),
          false
        );
      }
      break;

    default:
      cb(new Error("Invalid file type!"), false);
  }
};

const upload = multer({
  storage: memoryStorage,
  fileFilter,
  limits: {
    fileSize: 100 * 1024 * 1024,
  },
}).fields([
  { name: "AUDIO", maxCount: 1 },
  { name: "IMAGE", maxCount: 1 },
]);

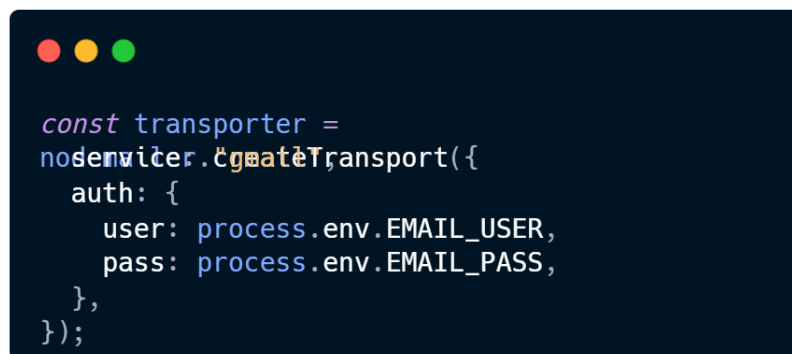
```

Gambar 15. *Input File Validation*

I. *Integration*

Penulis mengintegrasikan *third-party libraries* untuk meningkatkan fungsionalitas sistem, seperti:

- a. Nodemailer: Mengirim email notifikasi, termasuk tautan *reset password* untuk fitur pemulihan akun.



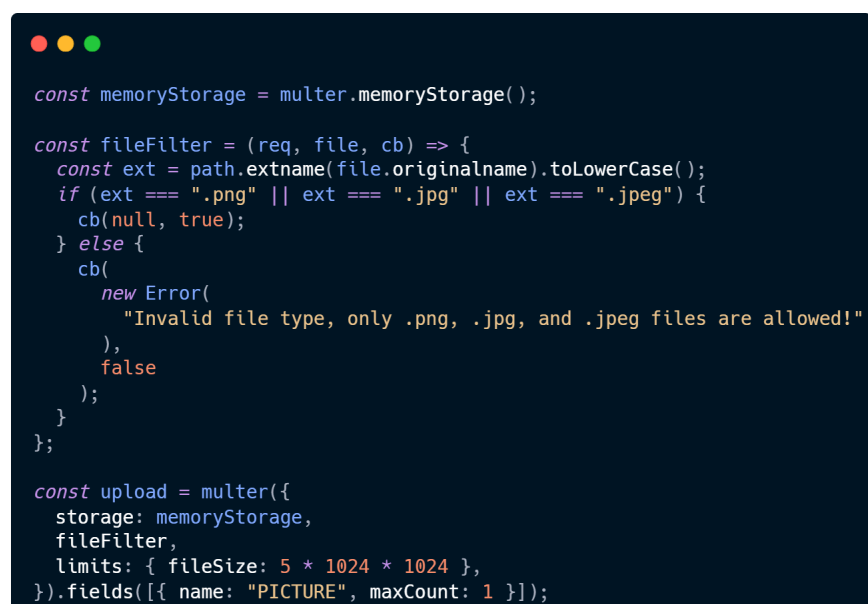
```

const transporter =
nodemailer.createTransport({
  auth: {
    user: process.env.EMAIL_USER,
    pass: process.env.EMAIL_PASS,
  },
});

```

Gambar 16. *Nodemailer Configuration*

- b. Multer: Mengelola unggahan *file* dengan efisien, mendukung berbagai jenis *media* seperti gambar dan *audio*.



```

const memoryStorage = multer.memoryStorage();

const fileFilter = (req, file, cb) => {
  const ext = path.extname(file.originalname).toLowerCase();
  if (ext === ".png" || ext === ".jpg" || ext === ".jpeg") {
    cb(null, true);
  } else {
    cb(
      new Error(
        "Invalid file type, only .png, .jpg, and .jpeg files are allowed!"
      ),
      false
    );
  }
};

const upload = multer({
  storage: memoryStorage,
  fileFilter,
  limits: { fileSize: 5 * 1024 * 1024 },
}).fields([ { name: "PICTURE", maxCount: 1 } ]);

```

Gambar 17. *Multer Configuration*

J. Dokumentasi *Endpoint Backend*

Pada bagian ini, akan menjelaskan mengenai endpoint yang ada pada aplikasi backend SEALS ini. Dibawah ini adalah penjelasan lebih mendetail mengenai endpoint-endpoint tersebut:

1. Auth

Pada Auth, berisi endpoint yang berkaitan dengan autentikasi user dalam sistem. Koleksi API ini mencakup fungsi seperti *register*, *login*, *validate email*, *forgot password*, dan *reset password*.

a. Register Student

Endpoint ini digunakan untuk melakukan register siswa ke sistem SEALS. Dibawah ini adalah route serta method yang digunakan.

Route	/api/register/student
Method	POST
Request	JSON raw

Tabel 1. *Route Register Student*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
NAME_USERS	body	String	Yes	Berisikan nama user yang akan didaftarkan.
EMAIL	body	String	Yes	Berisikan email untuk user yang sifatnya adalah unique (tidak boleh sama).
NISN	body	BigInt	Yes	Berisikan dari NISN user yang ingin didaftarkan.
PASSWORD	body	String	Yes	Berisikan password untuk user melakukan login nantinya.
CONFIRM_PASSWORD	body	String	Yes	Berisikan password sebelumnya untuk melakukan verifikasi

				agar tidak terjadi kesalahan penginputan password.
--	--	--	--	--

Tabel 2. *Parameter Register Student*

- Response

```
{
  "payload": null,
  "statusCode": 200,
  "message": "Student registered! Please verify your email."
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Student registered! Please verify your email.
400	Bad Request	a. {data} is required! b. {data} already registered!
500	Error Server	Internal Server Error.

Tabel 3. *Status Code Register Student*

b. Register Teacher

Endpoint ini digunakan untuk melakukan register guru ke sistem SEALS. Dibawah ini adalah route serta method yang digunakan.

Route	/api/register/teacher
Method	POST
Request	JSON raw

Tabel 4. *Route Register Teacher*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
-----------	------	-----------	----------	-------------

NAME_USERS	body	String	Yes	Berisikan nama user yang akan didaftarkan.
EMAIL	body	String	Yes	Berisikan email untuk user yang sifatnya adalah unique (tidak boleh sama).
NIP	body	BigInt	Yes	Berisikan dari NIP user yang ingin didaftarkan.
PASSWORD	body	String	Yes	Berisikan password untuk user melakukan login nantinya.
CONFIRM_PASSWORD	body	String	Yes	Berisikan password sebelumnya untuk melakukan verifikasi agar tidak terjadi kesalahan penginputan password.

Tabel 5. *Parameter Register Teacher*

- Response

```
{
  "payload": null,
  "statusCode": 200,
  "message": "Teacher registered! Please verify your email."
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Student registered! Please verify your email.
400	Bad Request	a. {data} is required! b. {data} already registered!
500	Error Server	Internal Server Error.

Tabel 6. *Status Code Register Teacher*

c. Validate Email

Endpoint ini digunakan untuk melakukan validasi email user yang dikirimkan ke sistem SEALS. Dibawah ini adalah route serta method yang digunakan.

Route	/api/validateEmail
Method	POST
Request	JSON raw

Tabel 7. *Route Validate Email*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
TOKEN	body	String	Yes	Berisikan token yang dikirimkan melalui email yang telah didaftarkan.

Tabel 8. *Parameter Validate Email*

- Response

```
{
  "payload": null,
  "statusCode": 200,
  "message": "Email successfully validated!"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Email successfully validated!
400	Bad Request	Invalid or expired token

Tabel 9. *Status Code Validate Email*

d. Login

Endpoint ini digunakan untuk melakukan login ke sistem SEALS. Dibawah ini adalah route serta method yang digunakan.

Route	/api/login
Method	POST

Request	JSON raw
---------	----------

Tabel 10. *Route Login*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
EMAIL	body	String	Yes	Berisikan email pengguna yang telah terdaftar pada sistem SEALS.
PASSWORD	body	String	Yes	Berisikan password pengguna.

Tabel 11. *Parameter Login*

- Response

```
{
  "payload": {
    "ID": "26350116-9451-4f56-9581-42493db2389d",
    "NAME_USERS": "Putra",
    "EMAIL": "putra@gmail.com",
    "ROLE": "student",
    "TOKEN": "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJJRRCi6IjI2MzUwMTE2LTk0NTUyOTI5ODUwLCJleHAiOiE3Mjc5NDA2NTB9.XHG-S76NXnimzOA3jZlgcMyxdhnuJON2QZCIE_20S9M",
    "REFRESH_TOKEN": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJJRRCi6IjI2MzUwMTE2LTk0NTUyOTI5ODUwLCJleHAiOiE3Mjc5NDA2NTB9.WsLjhb1NNrTXBxMXF6RN0u3ux8Pn8wyjRKJw5yytRQs"
  },
  "statusCode": 200,
  "message": "Login successful"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Login successful
400	Bad Request	{data} is required!
401	Unauthorized	The password you entered is incorrect!
403	Forbidden	User is not validated! Please verify your email first.
404	Not Found	User data not found!
500	Error Server	Internal Server Error.

Tabel 12. *Status Code Login*

e. Refresh Token

Endpoint ini digunakan untuk melakukan refresh token user yang dikirimkan ke sistem SEALS. Refresh token ini akan mengembalikan hak akses terhadap endpoint untuk user jika aksesnya telah expired. Dibawah ini adalah route serta method yang digunakan.

Route	/api/refreshToken
Method	POST
Request	JSON raw

Tabel 13. *Route Refresh Token*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
REFRESH_TOKEN	body/cookies	String	Yes	Berisikan token yang dikirimkan melalui payload untuk mobile dan cookies untuk website.

Tabel 14. *Parameter Refresh Token*

- Response

```
{
  "payload": {
    "TOKEN": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJJRCI6IjI2MzUwMTE2LTk0NTU0NTg3LTQyNDkzZGIyMzg5ZCIsIlJPTEUiOiJzdHVkZW50IiwiaWF0IjoxNzI3OTI4NDg3LCJleHAiOjE3Mjc5MzkyODd9.JQH7Xv5MtlPNO86KW1U92N28dWP8jcfRS5A-WHtHAjM",
    "REFRESH_TOKEN": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJJRCI6IjI2MzUwMTE2LTk0NTU0NTg3LTQyNDkzZGIyMzg5ZCIsIlJPTEUiOiJzdHVkZW50IiwiaWF0IjoxNzI3OTI4NDg3LCJleHAiOjE3Mjc5MzkyODd9.Z-SmCBV3wSZfdvbOotnD0D18KEyur17601K3JF8f-uM"
  },
  "statusCode": 200,
  "message": "Token refreshed successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Token refreshed successfully
400	Bad Request	Refresh token is required!
401	Unauthorized	Refresh token expired. Please login again.
403	Forbidden	Invalid refresh token!
500	Server Error	Internal Server Error

Tabel 15. *Status Code Refresh Token*

f. Forgot Password

Endpoint ini digunakan untuk melakukan perubahan password jika user mengalami lupa password dengan mengirimkan email user ke sistem SEALS. Dibawah ini adalah route serta method yang digunakan.

Route	/api/forgotPassword
Method	POST
Request	JSON raw

Tabel 16. *Route Forgot Password*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
EMAIL	body	String	Yes	Pengguna mengisi email pada endpoint ini dan akan dikirimkan token melalui email.

Tabel 17. *Parameter Forgot Password*

- Response

```
{
  "payload": null,
  "statusCode": 200,
  "message": "Password reset email sent successfully!"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Password reset email sent successfully!
400	Bad Request	Email is required!
404	Not Found	Email is not registered!
500	Server Error	Internal Server Error

Tabel 18. *Status Code Forgot Password*

g. Reset Password

Endpoint ini digunakan untuk melakukan perubahan password jika user mengalami lupa password. Dibawah ini adalah route serta method yang digunakan.

Route	/api/resetPassword
Method	POST
Request	JSON raw

Tabel 19. *Route Reset Password*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
TOKEN	body	String	Yes	Berisikan token yang didapatkan dari email yang dikirimkan ke pengguna.
NEW_PASSWORD	body	String	Yes	Password baru pengguna.
CONFIRM_NEW_PASSWORD	body	String	Yes	Konfirmasi password baru pengguna guna menghindari kesalahan input.

Tabel 20. *Parameter Reset Password*

- Response

```
{
  "payload": null,
  "statusCode": 200,
  "message": "Password has been reset successfully!"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Password has been reset successfully!
400	Bad Request	{data} is required!
404	Not Found	User data not found!
500	Server Error	Internal Server Error

Tabel 21. *Status Code Reset Password*

2. User Model

Pada User Model, berisi endpoint yang berkaitan dengan manipulasi data user dalam sistem. Koleksi API ini mencakup fungsi seperti *Get Users*, *Get User by Id*, *Get Me*, *Get Teachers*, *Get Students*, *Update User*, *Update User Password*, dan *Delete User*.

a. Get Users

Endpoint ini digunakan untuk mendapatkan seluruh data user yang ada (termasuk admin, teacher, dan student). Dibawah ini adalah route serta method yang digunakan.

Route	/api/user
Method	GET
Request	JSON raw

Tabel 22. *Route Get Users*

- Parameters

Pada endpoint ini, tidak perlu mengirimkan parameter apapun, cukup memanggil route beserta methodnya yang nantinya akan divalidasi oleh sistem backend.

- Response

```
{
  "payload": [
    {
      "ID": "1debf8aa-cc2b-435a-8c1c-8e67bca2e918",
      "NAME_USERS": "Naresh",
      "EMAIL": "nareshteach@gmail.com",
```

```

        "ROLE": "teacher",
        "PICTURE": "default-avatar.jpeg",
        "NIP/NISN": 2141720058
    },
    {
        "ID": "12027fb9-365f-485e-8b23-52e2bc3e4352",
        "NAME_USERS": "Adam",
        "EMAIL": "adam@gmail.com",
        "ROLE": "admin",
        "PICTURE": "default-avatar.jpeg",
        "NIP/NISN": null
    },
    {
        "ID": "5b8234cc-000f-4c77-ad3d-ab211cceeff1",
        "NAME_USERS": "elang",
        "EMAIL": "elangptra16@gmail.com",
        "ROLE": "student",
        "PICTURE": "default-avatar.jpeg",
        "NIP/NISN": 1345484845
    },
    {
        "ID": "7351d439-f42a-4184-b6d5-e05dbab08c85",
        "NAME_USERS": "elangptra",
        "EMAIL": "elangptra17@gmail.com",
        "ROLE": "student",
        "PICTURE": "default-avatar.jpeg",
        "NIP/NISN": 2141720077
    },
    {
        "ID": "7f8d9f28-b5c5-444a-9e12-718300303849",
        "NAME_USERS": "Elang",
        "EMAIL": "elang@gmail.com",
        "ROLE": "admin",
        "PICTURE": "default-avatar.jpeg",
        "NIP/NISN": null
    },
    ],
    "statusCode": 200,
    "message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Internal Server Error

Tabel 23. *Status Code Get Users*

b. Get User by id

Endpoint ini digunakan untuk mendapatkan data user yang ada berdasarkan id (termasuk admin, teacher, dan student). Dibawah ini adalah route serta method yang digunakan.

Route	/api/user/:id
Method	GET
Request	JSON raw

Tabel 24. *Route Get User by Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari user yang ingin di tampilkan.

Tabel 25. *Parameter Get User by Id*

- Response

```
{
  "payload": {
    "ID": "a50c09c2-289f-43b0-a6ef-6a98a8de1314",
    "NAME_USERS": "Maguire",
    "EMAIL": "maguire@gmail.com",
    "ROLE": "student",
    "NISN": 6419184864,
    "NAME_CLASS": null,
    "PICTURE": "user-4737cfedb947b53455365339ed73da58.jpeg"
  },
}
```

```

"statusCode": 200,
"message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
404	Not Found	User not found
500	Server Error	Internal Server Error

Tabel 26. Status Code Get User by Id

c. Get Me

Endpoint ini digunakan untuk mendapatkan data user yang login pada sistem saat ini (termasuk admin, teacher, dan student). Dibawah ini adalah route serta method yang digunakan.

Route	/api/getMe
Method	GET
Request	JSON raw

Tabel 27. Route Get Me

- Parameters

Pada endpoint ini, tidak perlu mengirimkan parameter apapun, cukup memanggil route beserta methodnya yang nantinya akan divalidasi oleh sistem backend.

- Response

```

{
  "payload": {
    "ID": "26350116-9451-4f56-9581-42493db2389d",
    "NAME_USERS": "Putra",
    "EMAIL": "putra@gmail.com",
    "ROLE": "student",
    "NISN": 2141720074,
    "NAME_CLASS": "Kelas A",
  }
}

```

```

    "PICTURE": "user-e7e557c7a17b1c5e642f7a679476820b.jpeg"
  },
  "statusCode": 200,
  "message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
404	Not Found	User not found
500	Server Error	Error retrieving user data!

Tabel 28. *Status Code Get Me*

d. Get Teachers

Endpoint ini digunakan untuk mendapatkan data user yang memiliki role teacher. Dibawah ini adalah route serta method yang digunakan.

Route	/api/user/teacher?page=&limit=&search=&sort=
Method	GET
Request	JSON raw

Tabel 29. *Route Get Teachers*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
page	params	String	No	Digunakan untuk berpindah page dari hasil get, nilai default adalah 1.
limit	params	String	No	Untuk menampilkan banyaknya data yang

				ditampilkan dalam 1 page, nilai default adalah 10.
search	params	String	No	Melakukan search data pada database.
sort	params	String	No	Melakukan sorting data pada database.

Tabel 30. *Parameter Get Teachers*

- Response

```
{
  "payload": {
    "teachers": [
      {
        "ID": "d9838d4c-b396-4687-9d48-b9cd91b35d56",
        "NAME_USERS": "Naresh",
        "EMAIL": "nareshteachz@gmail.com",
        "NIP": 2141720053,
        "ROLE": "teacher",
        "TIME_USERS": null
      },
      {
        "ID": "1debf8aa-cc2b-435a-8c1c-8e67bca2e918",
        "NAME_USERS": "Naresh",
        "EMAIL": "nareshteach@gmail.com",
        "NIP": "192039485968349583",
        "ROLE": "teacher",
        "TIME_USERS": null
      },
      {
        "ID": "cd2d58dc-dd04-4f7a-b56d-6d4f2698ddbc",
        "NAME_USERS": "Naresh",
        "EMAIL": "nareshteachs@gmail.com",
        "NIP": 2141720058,
        "ROLE": "teacher",
        "TIME_USERS": null
      }
    ],
    "currentPage": 1,
    "totalPages": 1,
    "totalSearchedItems": 3,
    "totalTeachers": 4
  }
}
```



```

},
"statusCode": 200,
"message": "Teachers retrieved successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Teachers retrieved successfully
500	Server Error	Error retrieving teacher data!

Tabel 31. *Status Code Get Teachers*

e. Get Students

Endpoint ini digunakan untuk mendapatkan data user yang memiliki role student. Dibawah ini adalah route serta method yang digunakan.

Route	/api/user/student?page=&limit=&search=&sort=
Method	GET
Request	JSON raw

Tabel 32. *Route Get Students*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
page	params	String	No	Digunakan untuk berpindah page dari hasil get, nilai default adalah 1.
limit	params	String	No	Untuk menampilkan banyaknya data yang ditampilkan dalam 1 page, nilai default adalah

				10.
search	params	String	No	Melakukan search data pada database.
sort	params	String	No	Melakukan sorting data pada database.

Tabel 33. *Parameter Get Students*

- Response

```
{
  "payload": {
    "students": [
      {
        "ID": "5b8234cc-000f-4c77-ad3d-ab211cceeef1",
        "NISN": 1345484845,
        "NAME_USERS": "elang",
        "EMAIL": "elangptra16@gmail.com",
        "NAME_CLASS": null,
        "ROLE": "student",
        "TIME_USERS": null
      },
      {
        "ID": "fee4e191-7eea-4503-a64a-c0f1b255c2d5",
        "NISN": 2141720073,
        "NAME_USERS": "elangs",
        "EMAIL": "elangs@gmail.com",
        "NAME_CLASS": "Kelas A",
        "ROLE": "student",
        "TIME_USERS": null
      },
      {
        "ID": "96f6a0c7-dcb1-4014-be14-85e94b107957",
        "NISN": 2141720076,
        "NAME_USERS": "elangs",
        "EMAIL": "elangsa@gmail.com",
        "NAME_CLASS": null,
        "ROLE": "student",
        "TIME_USERS": null
      },
      {
        "ID": "7351d439-f42a-4184-b6d5-e05dbab08c85",
        "NISN": 2141720077,
```

```

        "NAME_USERS": "elangptra",
        "EMAIL": "elangptra17@gmail.com",
        "NAME_CLASS": "Kelas A",
        "ROLE": "student",
        "TIME_USERS": null
    },
    {
        "ID": "8c3466f1-35cb-470f-aa84-16c52fd3db8e",
        "NISN": 214172012,
        "NAME_USERS": "elangsz",
        "EMAIL": "elangsz@gmail.com",
        "NAME_CLASS": null,
        "ROLE": "student",
        "TIME_USERS": null
    }
],
"currentPage": 1,
"totalPages": 2,
"totalSearchedItems": 6,
"totalStudents": 14
},
"statusCode": 200,
"message": "Students retrieved successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Teachers retrieved successfully
500	Server Error	Error retrieving teacher data!

Tabel 34. *Status Code Get Students*

f. Update User

Endpoint ini digunakan untuk melakukan update data user, baik itu admin, teacher, ataupun student (dengan catatan jika yang diupdate adalah admin, maka tidak perlu menambahkan field NIP/NISN). Dibawah ini adalah route serta method yang digunakan.

Route	/api/user/update/:id
Method	PUT
Request	form-data

Tabel 35. *Route Update User*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
NAME_USERS	body	String	No	Berisikan nama user yang akan didaftarkan.
EMAIL	body	String	No	Berisikan email untuk user yang sifatnya adalah unique (tidak boleh sama).
NIP/NISN	body	BigInt	No	Berisikan NIP/NISN user yang ingin diupdate (tidak perlu ditambahkan jika user tersebut adalah admin).
PICTURE	body	file	No	Berisikan foto profile pengguna yang ingin dilakukan update

Tabel 36. *Parameter Update User*

- Response

```
{
  "payload": {
    "ID": "a50c09c2-289f-43b0-a6ef-6a98a8de1314",
    "NAME_USERS": "Maguiresh",
    "EMAIL": "maguiresh@gmail.com",
    "ROLE": "student",
    "PICTURE": "user-d3b55e105e3f6aaae08de98c6d4fb98d.jpeg",
    "NISN": 6419184865
  },
  "statusCode": 200,
}
```

```

"message": "User updated successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	User updated successfully
400	Bad Request	{data} already registered!
404	Not Found	User not found
500	Error Server	Internal Server Error.

Tabel 37. Status Code Update User

g. Update User Password

Endpoint ini digunakan untuk melakukan update password user, baik itu admin, teacher, ataupun student. Dibawah ini adalah route serta method yang digunakan.

Route	/api/user/update/password/:id
Method	PUT
Request	form-data

Tabel 38. Route Update User Password

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	Berisikan id dari user yang ingin mengganti password.
OLD_PASSWORD	body	String	Yes	Berisikan password sebelumnya milik pengguna.

PASSWO RD	body	String	Yes	Berisikan password baru milik user yang ingin digunakan.
CONFIR M_PASS WORD	body	String	Yes	Berisikan password yang sebelumnya diinputkan oleh pengguna pada parameter PASSWORD.

Tabel 39. *Parameter Update User Password*

- Response

```
{
  "payload": null,
  "statusCode": 200,
  "message": "Password updated successfully."
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Password updated successfully.
400	Bad Request	{data} must be filled!
404	Not Found	User not found
500	Error Server	Internal Server Error.

Tabel 40. *Status Code Update User Password*

h. Delete User

Endpoint ini digunakan untuk melakukan delete data user dari sistem.

Dibawah ini adalah route serta method yang digunakan.

Route	/api/user/delete/:id
Method	DELETE
Request	JSON raw

Tabel 41. *Route Delete User*

- Parameters

Pada endpoint ini, perlu mengirimkan id user untuk melakukan delete data user.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	Berisikan id dari user yang ingin dihapus.

Tabel 42. *Parameter Delete User*

- Response

```
{
  "payload": null,
  "statusCode": 200,
  "message": "User deleted successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
404	Not Found	User not found
500	Server Error	Error retrieving user data!

Tabel 43. *Status Code Delete User*

3. Section Model

Pada Section Model, berisi endpoint yang berkaitan dengan manipulasi data section dalam sistem. Koleksi API ini mencakup fungsi seperti *Get Sections*, *Get Section by Id*, *Get Section For Admin Page*, *Create Section*, *Update Section*, dan *Delete Section*.

a. Get Sections

Endpoint ini digunakan untuk mendapatkan seluruh data section yang ada. Dibawah ini adalah route serta method yang digunakan.

Route	/api/section
-------	--------------

Method	GET
Request	JSON raw

Tabel 44. *Route Get Section*

- Parameters

Pada endpoint ini, tidak perlu mengirimkan parameter apapun, cukup memanggil route beserta methodnya yang nantinya akan divalidasi oleh sistem backend.

- Response

```
{
  "payload": [
    {
      "ID_SECTION": "49390272-efbd-4f1c-9b63-39fa8905b891",
      "NAME_SECTION": "Membaca",
      "DESCRIPTION_SECTION": "Membaca adalah membaca",
      "THUMBNAIL": "thumbnail-23e53ab0423f77bf1a8c354d183115bf.jpeg",
      "TIME_SECTION": "2024-08-26T00:03:48.000Z"
    },
    {
      "ID_SECTION": "a80dc3c9-f12b-4015-b33f-e1fce7df0abc",
      "NAME_SECTION": "Menulis",
      "DESCRIPTION_SECTION": "Menulis adalah menulis",
      "THUMBNAIL": "thumbnail-a80dc3c9-f12b-4015-b33f-e1fce7df0abc-dbef05e3259efe0c2cb073f6eb32dc14.png",
      "TIME_SECTION": "2024-08-27T20:34:28.000Z"
    }
  ],
  "statusCode": 200,
  "message": "Success"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Internal Server Error

Tabel 45. *Status Code Get Section*

b. Get Section by Id

Endpoint ini digunakan untuk mendapatkan data section yang ada berdasarkan id. Dibawah ini adalah route serta method yang digunakan.

Route	/api/section/:id
Method	GET
Request	JSON raw

Tabel 46. *Route Get Section by Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari section yang ingin di tampilkan.

Tabel 47. *Parameter Get Section by Id*

- Response

```
{
  "payload": {
    "ID_SECTION": "a80dc3c9-f12b-4015-b33f-e1fce7df0abc",
    "NAME_SECTION": "Menulis",
    "DESCRIPTION_SECTION": "Menulis adalah menulis",
    "THUMBNAIL":
    "thumbnail-a80dc3c9-f12b-4015-b33f-e1fce7df0abc-dbef05e3259efe0c2cb073f6eb32dc14.png",
    "TIME_SECTION": "2024-08-27T20:34:28.000Z"
  },
  "statusCode": 200,
  "message": "Success"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
404	Not Found	Section not found
500	Server Error	Internal Server Error

Tabel 48. *Status Code Get Section by Id*

c. *get Section For Admin Page*

Endpoint ini digunakan untuk mendapatkan seluruh data section yang ada untuk halaman admin dengan beberapa parameter untuk pagination, search, dan sort. Dibawah ini adalah route serta method yang digunakan.

Route	/api/section/admin?page=&limit=&search=&sort=
Method	GET
Request	JSON raw

Tabel 49. *Route Get Section For Admin Page*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
page	params	int	No	page yang ingin ditampilkan.
limit	params	int	No	limit dari banyaknya data yang akan ditampilkan pada sebuah page (default adalah 10).
search	params	String	No	parameter untuk melakukan search data section.
sort	params	String	No	parameter untuk melakukan sort.

Tabel 50. *Parameter Get Section For Admin Page*

- Response

```
{
  "payload": {
    "sections": [
      {
        "ID_SECTION": "3e9cff3b-42ea-4c51-ab2f-836a7e6adf50",
        "NAME_SECTION": "Grammar",
        "DESCRIPTION_SECTION": "English Adaptive Learning (Grammar Section)",
        "THUMBNAIL": "THUMBNAIL-3e9cff3b-42ea-4c51-ab2f-836a7e6adf50-dbef05e3259efe0c2cb073f6eb32dc14.png",
        "TIME_SECTION": "2024-09-19T03:26:55.000Z"
      },
      {
        "ID_SECTION": "49390272-efbd-4f1c-9b63-39fa8905b891",
        "NAME_SECTION": "Listening",
        "DESCRIPTION_SECTION": "English Adaptive Learning (Listening Section)",
        "THUMBNAIL": "THUMBNAIL-49390272-efbd-4f1c-9b63-39fa8905b891-3b4db19865d8b70386db0834852659ac.jpeg",
        "TIME_SECTION": "2024-08-26T00:03:48.000Z"
      },
      {
        "ID_SECTION": "39e8b29b-e6d6-4e63-b597-c012896c67b7",
        "NAME_SECTION": "Reading",
        "DESCRIPTION_SECTION": "English Adaptive Learning (Reading Section)",
        "THUMBNAIL": "THUMBNAIL-3e9cff3b-42ea-4c51-ab2f-836a7e6adf50-dbef05e3259efe0c2cb073f6eb32dc14.png",
        "TIME_SECTION": "2024-09-19T03:07:05.000Z"
      }
    ],
    "currentPage": 1,
    "totalPages": 1,
    "totalItems": 3
  },
  "statusCode": 200,
  "message": "Sections retrieved successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Internal Server Error

Tabel 51. *Status Code Get Section For Admin Page*

d. Create Section

Endpoint ini digunakan untuk membuat sebuah data section pada sistem.

Dibawah ini adalah route serta method yang digunakan.

Route	/api/section
Method	POST
Request	form-data

Tabel 52. *Route Create Section*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
NAME_SECTION	body	String	Yes	Berisikan nama dari section yang ingin dibuat.
DESCRIPTION_SECTION	body	String	Yes	Deskripsi section yang akan dibuat.
THUMB NAIL	body	file	No	Thumbnail dari section yang dibuat.

Tabel 53. *Parameter Create Section*

- Response

```
{
  "payload": {
    "ID_SECTION": "3e9cff3b-42ea-4c51-ab2f-836a7e6adf50",
    "TIME_SECTION": "2024-09-19T03:26:55.792Z",
    "NAME_SECTION": "Grammar",
    "DESCRIPTION_SECTION": "English Adaptive Learning
(Grammar Section)",
    "THUMBNAIL": "THUMBNAIL-3e9cff3b-42ea-4c51-ab2f-836a7e6adf50-dbef05e3259efe0
c2cb073f6eb32dc14.png"
  },
  "statusCode": 201,
  "message": "Section created successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
201	Created	Section created successfully
400	Bad Request	{data} is required
409	Conflict	Section already exists
500	Server Error	Internal Server Error

Tabel 54. Status Code Create Section

e. Update Section

Endpoint ini digunakan untuk melakukan update sebuah data section pada sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/section/update/:id
Method	PUT
Request	form-data

Tabel 55. Route Update Section

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari section yang ingin diupdate.
NAME_SECTION	body	String	Yes	Berisikan nama dari section yang ingin diupdate.
DESCRIPTION_SECTION	body	String	Yes	Deskripsi section yang akan diupdate.
THUMB NAIL	body	file	No	Thumbnail dari section yang diupdate.

Tabel 56. *Parameter Update Section*

- Response

```
{
  "payload": {
    "ID_SECTION": "49390272-efbd-4f1c-9b63-39fa8905b891",
    "NAME_SECTION": "Listening",
    "DESCRIPTION_SECTION": "English Adaptive Learning (Listening Section)",
    "THUMB NAIL": "THUMB NAIL-49390272-efbd-4f1c-9b63-39fa8905b891-81346cbd4317524acfd617bf14c006ce.jpeg",
    "TIME_SECTION": "2024-08-26T00:03:48.000Z"
  },
  "statusCode": 200,
  "message": "Section updated successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Section updated successfully
400	Bad Request	Section name already exists
404	Not Found	Section not found
500	Server Error	Internal Server Error

Tabel 57. *Status Code Update Section*

f. Delete Section

Endpoint ini digunakan untuk melakukan delete data section yang ada pada sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/section/delete/:id
Method	DELETE
Request	JSON raw

Tabel 58. *Route Delete Section*

- Parameters

Pada endpoint ini, perlu mengirimkan id user untuk melakukan delete data user.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	Berisikan id dari section yang ingin dihapus.

Tabel 59. *Parameter Delete Section*

- Response

```
{
  "payload": null,
  "statusCode": 200,
  "message": "Section deleted successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Section, topics, levels, and related exercises soft deleted successfully
404	Not Found	Section not found

500	Server Error	Internal Server Error
-----	--------------	-----------------------

Tabel 60. *Status Code Delete Section*

4. Topic Model

Pada Topic Model, berisi endpoint yang berkaitan dengan manipulasi data topic dalam sistem. Koleksi API ini mencakup fungsi seperti *Get Topics*, *Get Topic by Id*, *Get Topic By Section Id*, *Get Topic For Admin Page*, *Get Completed Topics*, *Create Topic*, *Update Topic*, dan *Delete Topic*.

a. Get Topics

Endpoint ini digunakan untuk mendapatkan seluruh data topic yang ada. Dibawah ini adalah route serta method yang digunakan.

Route	/api/topic
Method	GET
Request	JSON raw

Tabel 61. *Route Get Topics*

- Parameters

Pada endpoint ini, tidak perlu mengirimkan parameter apapun, cukup memanggil route beserta methodnya yang nantinya akan divalidasi oleh sistem backend.

- Response

```
{
  "payload": [
    {
      "ID_TOPIC": "11ccca8b-9ec4-4302-b2ad-1705d6650133",
      "ID_SECTION": "49390272-efbd-4f1c-9b63-39fa8905b891",
      "NAME_TOPIC": "Naresh Pratista",
      "DESCRIPTION_TOPIC": "Topic for Testing",
      "IS_DELETED": 0,
      "TIME_TOPIC": "2024-10-10T01:56:55.000Z"
    },
    {
      "ID_TOPIC": "19b625b0-9919-4761-ac07-679f942788da",
```



```

"ID_SECTION":
"39e8b29b-e6d6-4e63-b597-c012896c67b7",
  "NAME_TOPIC": "Naresh Prat",
  "DESCRIPTION_TOPIC": "Topic for Testing",
  "IS_DELETED": 0,
  "TIME_TOPIC": "2024-11-04T04:15:21.000Z"
},
{
  "ID_TOPIC": "21a9c23e-9737-49f5-bbd5-5b0ecac3f73d",
  "ID_SECTION":
"39e8b29b-e6d6-4e63-b597-c012896c67b7",
  "NAME_TOPIC": "The Legend of Naresh",
  "DESCRIPTION_TOPIC": "Story about Naresh going to
the West",
  "IS_DELETED": 0,
  "TIME_TOPIC": "2024-09-19T03:17:00.000Z"
},
{
  "ID_TOPIC": "8622dfea-83f3-4a43-aa8f-144762ff882a",
  "ID_SECTION":
"39e8b29b-e6d6-4e63-b597-c012896c67b7",
  "NAME_TOPIC": "Pratista the Hunter",
  "DESCRIPTION_TOPIC": "Journey of Pratista Hunting
the Monkey King",
  "IS_DELETED": 0,
  "TIME_TOPIC": "2024-09-19T03:15:58.000Z"
},
],
"statusCode": 200,
"message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Internal Server Error

Tabel 62. Status Code Get Topics

b. Get Topic by Id

Endpoint ini digunakan untuk mendapatkan data topic yang ada berdasarkan id. Dibawah ini adalah route serta method yang digunakan.

Route	/api/topic/:id
Method	GET
Request	JSON raw

Tabel 62. Route Get Topic by Id

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari topic yang ingin di tampilkan.

Tabel 63. Parameter Get Topic by Id

- Response

```
{
  "payload": {
    "ID_TOPIC": "21a9c23e-9737-49f5-bbd5-5b0ecac3f73d",
    "ID_SECTION": "39e8b29b-e6d6-4e63-b597-c012896c67b7",
    "NAME_TOPIC": "The Legend of Naresh",
    "DESCRIPTION_TOPIC": "Story about Naresh going to the
West",
    "IS_DELETED": 0,
    "TIME_TOPIC": "2024-09-19T03:17:00.000Z"
  },
  "statusCode": 200,
  "message": "Success"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success

404	Not Found	Topic not found
500	Server Error	Internal Server Error

Tabel 64. *Status Code Get Topic by Id*

c. Get Topic by Section Id

Endpoint ini digunakan untuk mendapatkan data topic yang berkaitan dengan sebuah section. Dibawah ini adalah route serta method yang digunakan.

Route	/api/topic/section/:sectionId
Method	GET
Request	JSON raw

Tabel 65. *Route Get Topic by Section Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
sectionId	params	String	Yes	id dari section untuk menampilkan list topic dari section tersebut.

Tabel 66. *Parameter Get Topic by Section Id*

- Response

```
{
  "payload": [
    {
      "ID_TOPIC": "11ccca8b-9ec4-4302-b2ad-1705d6650133",
      "NAME_TOPIC": "Naresh Pratista",
      "DESCRIPTION_TOPIC": "Topic for Testing",
      "IS_COMPLETED": 0
    },
    {
      "ID_TOPIC": "c2453b27-c9df-4f24-8502-ae32bd8044ad",
      "NAME_TOPIC": "Belajar Memahami Naresh",

```

```

        "DESCRIPTION_TOPIC": "Mempelajari tentang kehidupan
naresh",
        "IS_COMPLETED": 1
    }
],
"statusCode": 200,
"message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
404	Not Found	Section not found
500	Server Error	Internal Server Error

Tabel 67. Status Code Get Topic by Section Id

d. Get Topic For Admin Page

Endpoint ini digunakan untuk mendapatkan seluruh data topic yang digunakan pada halaman admin serta memiliki fitur seperti search, sort, dan pagination. Dibawah ini adalah route serta method yang digunakan.

Route	/api/topic/admin?page=&limit=&search=&sort=
Method	GET
Request	JSON raw

Tabel 68. Route Get Topic For Admin Page

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
page	params	int	No	page yang ingin ditampilkan.

limit	params	int	No	limit dari banyaknya data yang akan ditampilkan pada sebuah page (default adalah 10)
search	params	String	No	parameter untuk melakukan search data topic
sort	params	String	No	parameter untuk melakukan sort.

Tabel 69. *Parameter Get Topic For Admin Page*

- Response

```
{
  "payload": {
    "topics": [
      {
        "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
        "ID_TOPIC":
"c2453b27-c9df-4f24-8502-ae32bd8044ad",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Naresh",
        "DESCRIPTION_TOPIC": "Mempelajari tentang
kehidupan naresh",
        "TIME_TOPIC": "2024-08-26T00:50:21.000Z"
      },
      {
        "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
        "ID_TOPIC":
"891f6b2e-b385-4dc5-9bbd-97a44daf7138",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Pratista",
        "DESCRIPTION_TOPIC": "Mempelajari tentang
kehidupan pratista",
        "TIME_TOPIC": "2024-08-29T07:52:44.000Z"
      },
      {
        "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
        "ID_TOPIC":
"11ccca8b-9ec4-4302-b2ad-1705d6650133",
```

```

        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Naresh Pratista",
        "DESCRIPTION_TOPIC": "Topic for Testing",
        "TIME_TOPIC": "2024-10-10T01:56:55.000Z"
    },
    ],
    "currentPage": 1,
    "totalPages": 1,
    "totalItems": 3
},
"statusCode": 200,
"message": "Topics retrieved successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Internal Server Error

Tabel 70. *Status Code Get Topic For Admin Page*

e. Get Completed Topics

Endpoint ini digunakan untuk mendapatkan seluruh data topic yang telah diselesaikan oleh pengguna login saat ini. Dibawah ini adalah route serta method yang digunakan.

Route	/api/topic/complete
Method	GET
Request	JSON raw

Tabel 71. *Route Get Completed Topics*

- Parameters

Pada endpoint ini, tidak perlu mengirimkan parameter apapun, cukup memanggil route beserta methodnya yang nantinya akan divalidasi oleh sistem backend.

- Response

```
{
  "payload": [
    {
      "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
      "NAME_SECTION": "Listening",
      "DESCRIPTION_SECTION": "English Adaptive Learning
(Listening Section)",
      "THUMBNAIL":
"THUMBNAIL-49390272-efbd-4f1c-9b63-39fa8905b891-3b4db19865d8b70
386db0834852659ac.jpeg",
      "TOTAL_TOPICS": 3,
      "COMPLETED_TOPICS": 2
    }
  ],
  "statusCode": 200,
  "message": "Completed topics by section fetched
successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Completed topics by section fetched successfully
500	Server Error	Internal Server Error

Tabel 72. *Status Code Get Completed Topics*

f. Create Topic

Endpoint ini digunakan untuk membuat sebuah data topic pada sistem yang akan langsung membuat data level mulai dari pretest hingga level 6 secara otomatis. Dibawah ini adalah route serta method yang digunakan.

Route	/api/topic
Method	POST
Request	JSON raw

Tabel 73. *Route Create Topic*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
ID_SECTION	body	String	Yes	Memuat id section yang ingin dikaitkan dengan topic yang dibuat.
NAME_TOPIC	body	String	Yes	Nama topic yang akan dibuat.
DESCRIPTION_TOPIC	body	String	Yes	Deskripsi dari topic yang akan dibuat.

Tabel 74. *Parameter Create Topic*

- Response

```
{
  "payload": {
    "topic": {
      "ID_TOPIC": "d3880896-7e19-4c2e-8d2b-68ba90c6a3e5",
      "IS_DELETED": 0,
      "TIME_TOPIC": "2024-10-16T03:11:35.528Z",
      "ID_SECTION": "49390272-efbd-4f1c-9b63-39fa8905b891",
      "NAME_TOPIC": "Pratista Naresh",
      "DESCRIPTION_TOPIC": "Topic for Testing"
    },
    "levels": [
      {
        "ID_LEVEL": "9ec21f58-d963-4611-aa8b-c8103c3b078d",
        "ID_TOPIC": "d3880896-7e19-4c2e-8d2b-68ba90c6a3e5",
        "ID_SECTION": "49390272-efbd-4f1c-9b63-39fa8905b891",
        "NAME_LEVEL": "Level 1",
        "CONTENT": null,
        "AUDIO": null,
        "IMAGE": null,
        "VIDEO": null,
        "IS_PRETEST": 0,

```



```

"ROUTE_1":
"6733fc4f-d6eb-4a02-a60b-27cb2fb9995b",
"ROUTE_2":
"b34a831d-fb1d-4811-9c7d-a440a1742769",
"ROUTE_3":
"7cd99a6c-4058-42d8-ba3d-b0fef17a5f5f",
"ROUTE_4":
"5c0ebd03-3b2b-448e-9e80-bc646807cc82",
"ROUTE_5":
"7ff509c7-5977-4b9b-8cfe-653ea358edee",
"ROUTE_6":
"2b32b08c-ab15-4a97-b410-50a0557a057f",
    "IS_DELETED": 0,
    "TIME_LEVEL": "2024-10-16T03:11:35.000Z"
  },
  {
    "ID_LEVEL":
"b34a831d-fb1d-4811-9c7d-a440a1742769",
    "ID_TOPIC":
"d3880896-7e19-4c2e-8d2b-68ba90c6a3e5",
    "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
    "NAME_LEVEL": "Level 2",
    "CONTENT": null,
    "AUDIO": null,
    "IMAGE": null,
    "VIDEO": null,
    "IS_PRETEST": 0,
    "ROUTE_1":
"6733fc4f-d6eb-4a02-a60b-27cb2fb9995b",
    "ROUTE_2":
"9ec21f58-d963-4611-aa8b-c8103c3b078d",
    "ROUTE_3":
"7cd99a6c-4058-42d8-ba3d-b0fef17a5f5f",
    "ROUTE_4":
"5c0ebd03-3b2b-448e-9e80-bc646807cc82",
    "ROUTE_5":
"7ff509c7-5977-4b9b-8cfe-653ea358edee",
    "ROUTE_6":
"2b32b08c-ab15-4a97-b410-50a0557a057f",
    "IS_DELETED": 0,
    "TIME_LEVEL": "2024-10-16T03:11:35.000Z"
  },
  {

```

```

"ID_LEVEL":
"7cd99a6c-4058-42d8-ba3d-b0fef17a5f5f",
"ID_TOPIC":
"d3880896-7e19-4c2e-8d2b-68ba90c6a3e5",
"ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
      "NAME_LEVEL": "Level 3",
      "CONTENT": null,
      "AUDIO": null,
      "IMAGE": null,
      "VIDEO": null,
      "IS_PRETEST": 0,
      "ROUTE_1":
"6733fc4f-d6eb-4a02-a60b-27cb2fb9995b",
      "ROUTE_2":
"9ec21f58-d963-4611-aa8b-c8103c3b078d",
      "ROUTE_3":
"b34a831d-fb1d-4811-9c7d-a440a1742769",
      "ROUTE_4":
"5c0ebd03-3b2b-448e-9e80-bc646807cc82",
      "ROUTE_5":
"7ff509c7-5977-4b9b-8cfe-653ea358edee",
      "ROUTE_6":
"2b32b08c-ab15-4a97-b410-50a0557a057f",
      "IS_DELETED": 0,
      "TIME_LEVEL": "2024-10-16T03:11:35.000Z"
    },
    {
      "ID_LEVEL":
"5c0ebd03-3b2b-448e-9e80-bc646807cc82",
      "ID_TOPIC":
"d3880896-7e19-4c2e-8d2b-68ba90c6a3e5",
      "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
      "NAME_LEVEL": "Level 4",
      "CONTENT": null,
      "AUDIO": null,
      "IMAGE": null,
      "VIDEO": null,
      "IS_PRETEST": 0,
      "ROUTE_1":
"6733fc4f-d6eb-4a02-a60b-27cb2fb9995b",
      "ROUTE_2":
"9ec21f58-d963-4611-aa8b-c8103c3b078d",

```

```

"ROUTE_3":
"b34a831d-fb1d-4811-9c7d-a440a1742769",
"ROUTE_4":
"7cd99a6c-4058-42d8-ba3d-b0fef17a5f5f",
"ROUTE_5":
"7ff509c7-5977-4b9b-8cfe-653ea358edee",
"ROUTE_6":
"2b32b08c-ab15-4a97-b410-50a0557a057f",
    "IS_DELETED": 0,
    "TIME_LEVEL": "2024-10-16T03:11:35.000Z"
  },
  {
    "ID_LEVEL":
"7ff509c7-5977-4b9b-8cfe-653ea358edee",
    "ID_TOPIC":
"d3880896-7e19-4c2e-8d2b-68ba90c6a3e5",
    "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
    "NAME_LEVEL": "Level 5",
    "CONTENT": null,
    "AUDIO": null,
    "IMAGE": null,
    "VIDEO": null,
    "IS_PRETEST": 0,
    "ROUTE_1":
"6733fc4f-d6eb-4a02-a60b-27cb2fb9995b",
    "ROUTE_2":
"9ec21f58-d963-4611-aa8b-c8103c3b078d",
    "ROUTE_3":
"b34a831d-fb1d-4811-9c7d-a440a1742769",
    "ROUTE_4":
"7cd99a6c-4058-42d8-ba3d-b0fef17a5f5f",
    "ROUTE_5":
"5c0ebd03-3b2b-448e-9e80-bc646807cc82",
    "ROUTE_6":
"2b32b08c-ab15-4a97-b410-50a0557a057f",
    "IS_DELETED": 0,
    "TIME_LEVEL": "2024-10-16T03:11:35.000Z"
  },
  {
    "ID_LEVEL":
"2b32b08c-ab15-4a97-b410-50a0557a057f",
    "ID_TOPIC":
"d3880896-7e19-4c2e-8d2b-68ba90c6a3e5",

```

```

"ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
      "NAME_LEVEL": "Level 6",
      "CONTENT": null,
      "AUDIO": null,
      "IMAGE": null,
      "VIDEO": null,
      "IS_PRETEST": 0,
      "ROUTE_1":
"6733fc4f-d6eb-4a02-a60b-27cb2fb9995b",
      "ROUTE_2":
"9ec21f58-d963-4611-aa8b-c8103c3b078d",
      "ROUTE_3":
"b34a831d-fb1d-4811-9c7d-a440a1742769",
      "ROUTE_4":
"7cd99a6c-4058-42d8-ba3d-b0fef17a5f5f",
      "ROUTE_5":
"5c0ebd03-3b2b-448e-9e80-bc646807cc82",
      "ROUTE_6":
"7ff509c7-5977-4b9b-8cfe-653ea358edee",
      "IS_DELETED": 0,
      "TIME_LEVEL": "2024-10-16T03:11:35.000Z"
    },
    {
      "ID_LEVEL":
"6733fc4f-d6eb-4a02-a60b-27cb2fb9995b",
      "ID_TOPIC":
"d3880896-7e19-4c2e-8d2b-68ba90c6a3e5",
      "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
      "NAME_LEVEL": "Pretest",
      "CONTENT": null,
      "AUDIO": null,
      "IMAGE": null,
      "VIDEO": null,
      "IS_PRETEST": 1,
      "ROUTE_1":
"9ec21f58-d963-4611-aa8b-c8103c3b078d",
      "ROUTE_2":
"b34a831d-fb1d-4811-9c7d-a440a1742769",
      "ROUTE_3":
"7cd99a6c-4058-42d8-ba3d-b0fef17a5f5f",
      "ROUTE_4":
"5c0ebd03-3b2b-448e-9e80-bc646807cc82",

```

```

"ROUTE_5":
"7ff509c7-5977-4b9b-8cfe-653ea358edee",
"ROUTE_6":
"2b32b08c-ab15-4a97-b410-50a0557a057f",
    "IS_DELETED": 0,
    "TIME_LEVEL": "2024-10-16T03:11:35.000Z"
  }
]
},
"statusCode": 201,
"message": "Topic and related Level created successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
201	Created	Topic and related Level created successfully
400	Bad Request	{data} is required
404	Not Found	Section not found
409	Conflict	Topic with the same name already exists
500	Server Error	Internal Server Error

Tabel 75. Status Code Create Topic

g. Update Topic

Endpoint ini digunakan untuk melakukan update sebuah data topic pada sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/topic/:id
Method	PUT
Request	JSON raw

Tabel 76. Route Update Topic

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
ID_SECTION	body	String	Yes	Memuat id section yang ingin dikaitkan dengan topic yang diupdate.
NAME_TOPIC	body	String	Yes	Nama topic yang akan diupdate.
DESCRIPTION_TOPIC	body	String	Yes	Deskripsi dari topic yang akan diupdate.

Tabel 77. *Parameter Update Topic*

- Response

```
{
  "payload": {
    "ID_TOPIC": "8622dfea-83f3-4a43-aa8f-144762ff882a",
    "ID_SECTION": "39e8b29b-e6d6-4e63-b597-c012896c67b7",
    "NAME_TOPIC": "Pratista the Hunter",
    "DESCRIPTION_TOPIC": "Journey of Pratista Hunting the
Monkey King",
    "TIME_TOPIC": "2024-09-19T03:15:58.000Z"
  },
  "statusCode": 200,
  "message": "Topic updated successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Topic updated successfully
404	Not Found	{data} not found
500	Server Error	Internal Server Error

Tabel 78. *Status Code Update Topic*

h. Delete Topic

Endpoint ini digunakan untuk melakukan delete data topic yang ada pada sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/topic/:id
Method	DELETE
Request	JSON raw

Tabel 79. *Route Delete Topic*

- Parameters

Pada endpoint ini, perlu mengirimkan id user untuk melakukan delete data user.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	Berisikan id dari topic yang ingin dihapus.

Tabel 80. *Parameter Delete Topic*

- Response

```
{
  "payload": null,
  "statusCode": 200,
  "message": "Topic deleted successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Topic, levels, and related exercises soft deleted successfully
404	Not Found	Topic not found
500	Server Error	Internal Server Error

Tabel 81. *Status Code Delete Topic*

5. Level Model

Pada Level Model, berisi endpoint yang berkaitan dengan manipulasi data level dalam sistem. Koleksi API ini mencakup fungsi seperti *Get Levels*, *Get Level by Id*, *Get Level By Topic Id*, *Get Level For Admin Page*, *Get Level Files*, *Update Level*, *Upload Level File*, *Delete Level File by Id and Levelname* dan *Delete Level*.

a. Get Levels

Endpoint ini digunakan untuk mendapatkan seluruh data level yang ada. Dibawah ini adalah route serta method yang digunakan.

Route	/api/level
Method	GET
Request	JSON raw

Tabel 82. Route *Get Levels*

- Parameters

Pada endpoint ini, tidak perlu mengirimkan parameter apapun, cukup memanggil route beserta methodnya yang nantinya akan divalidasi oleh sistem backend.

- Response

```
{
  "payload": [
    {
      "ID_LEVEL": "09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878",
      "ID_TOPIC": "891f6b2e-b385-4dc5-9bbd-97a44daf7138",
      "ID_SECTION": "49390272-efbd-4f1c-9b63-39fa8905b891",
      "NAME_LEVEL": "Pretest",
      "CONTENT": "Ini tes bang",
      "AUDIO": "audio-09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878-6174deec821539477b29c0b1aa6ff375.mp3",
      "IMAGE": null,
      "VIDEO": null,
      "IS_PRETEST": 1,
      "IS_DELETED": 0,
      "TIME_LEVEL": "2024-09-04T02:06:47.000Z"
    },
    {
      "ID_LEVEL": "0f89df44-edf8-40f6-a81b-1bc77628d798",
```



```

        "ID_TOPIC": "891f6b2e-b385-4dc5-9bbd-97a44daf7138",
        "ID_SECTION": "49390272-efbd-4f1c-9b63-39fa8905b891",
        "NAME_LEVEL": "Level 6",
        "CONTENT": "Ini tes bang",
        "AUDIO": null,
        "IMAGE":
        "image-0f89df44-edf8-40f6-a81b-1bc77628d798-876b42211a481cce1c3
a36b42979f3a5.png",
        "VIDEO": null,
        "IS_PRETEST": 0,
        "IS_DELETED": 0,
        "TIME_LEVEL": "2024-09-04T02:22:43.000Z"
    },
    "statusCode": 200,
    "message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Internal Server Error

Tabel 83. *Status Code Get Levels*

b. Get Level by Id

Endpoint ini digunakan untuk mendapatkan data level yang ada berdasarkan id. Dibawah ini adalah route serta method yang digunakan.

Route	/api/level/:id
Method	GET
Request	JSON raw

Tabel 84. *Route Get Level by Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari level yang ingin di tampilkan.

Tabel 85. *Parameter Get Level by Id*

- Response

```
{
  "payload": {
    "NAME_SECTION": "Reading",
    "NAME_TOPIC": "The Legend of Naresh",
    "ID_LEVEL": "17bad807-45c3-44e6-b1b5-8bd52fa22079",
    "ID_TOPIC": "21a9c23e-9737-49f5-bbd5-5b0ecac3f73d",
    "ID_SECTION": "39e8b29b-e6d6-4e63-b597-c012896c67b7",
    "NAME_LEVEL": "Pretest",
    "CONTENT": "In this pretest phase, students will assess their basic understanding of the main character, Naresh, and his journey to the West. The questions focus on the foundational elements of the story, such as Naresh's background, his motivation, and the introduction of some supporting characters. The goal is to measure the students' initial knowledge before moving on to the next levels.",
    "AUDIO": "AUDIO-17bad807-45c3-44e6-b1b5-8bd52fa22079-d9937102a559fbce6d6c55d66e799066.mp3",
    "IMAGE": "IMAGE-17bad807-45c3-44e6-b1b5-8bd52fa22079-f19cfaa031a896ebf1e3a3c28c18ad0c.jpeg",
    "VIDEO": "https://youtu.be/spMmq7j44A?si=ayVjBvoDawel0o3h",
    "IS_PRETEST": 1,
    "IS_DELETED": 0,
    "TIME_LEVEL": "2024-09-19T03:38:35.000Z"
  },
  "statusCode": 200,
  "message": "Success"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
------	-------	---------

200	Ok	Success
404	Not Found	Level not found
500	Server Error	Internal Server Error

Tabel 86. *Status Code Get Level by Id*

c. Get Level By Topic Id

Endpoint ini digunakan untuk mendapatkan data level yang ada berdasarkan dari topic id yang diberikan. Dibawah ini adalah route serta method yang digunakan.

Route	/api/level/topic/:idTopic
Method	GET
Request	JSON raw

Tabel 87. *Route Get Level by Topic Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
idTopic	params	String	Yes	id dari topic yang ingin di tampilkan level yang terikat dengan topic tersebut.

Tabel 88. *Parameter Get Level by Topic Id*

- Response

```
{
  "message": "Success",
  "data": {
    "lastCompletedLevel": {
      "ID_STUDENT_LEARNING":
"6b81668b-42e2-4254-af44-74063ac61ece",
      "ID_LEVEL": "0f89df44-edf8-40f6-a81b-1bc77628d798",
      "NAME_LEVEL": "Level 6",

```

```

"SCORE": 94,
"NEXT_LEARNING":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
"NEXT_LEARNING_LEVEL": "Level 6",
"FINISHED_AT": "2024-10-09T08:18:24.000Z",
"UNLOCKED_LEVELS": [
    "Level 1",
    "Level 2",
    "Level 3",
    "Level 4",
    "Level 5",
    "Pretest",
    "Level 6"
]
},
"levels": [
    {
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Pratista",
        "ID_LEVEL":
"09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878",
        "ID_TOPIC":
"891f6b2e-b385-4dc5-9bbd-97a44daf7138",
        "ID_SECTION":
"49390272-efbd-4flc-9b63-39fa8905b891",
        "NAME_LEVEL": "Pretest",
        "CONTENT": "Ini tes bang",
        "AUDIO":
"audio-09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878-6174deec821539477b2
9c0b1aa6ff375.mp3",
        "IMAGE": null,
        "VIDEO": null,
        "IS_PRETEST": 1,
        "IS_DELETED": 0,
        "TIME_LEVEL": "2024-09-04T02:06:47.000Z",
        "ID_STUDENT_LEARNING": "12341",
        "SCORE": 95
    },
    {
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Pratista",
        "ID_LEVEL":
"157216c2-0b33-440c-8782-c23e4dfc4af3",

```

```

"ID_TOPIC":
"891f6b2e-b385-4dc5-9bbd-97a44daf7138",
"ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
  "NAME_LEVEL": "Level 1",
  "CONTENT": "testing level 1",
  "AUDIO": null,
  "IMAGE":
"IMAGE-157216c2-0b33-440c-8782-c23e4dfc4af3-9b7abd31eb795a753c6
54509ff2357ea.jpeg",
  "VIDEO":
"https://youtu.be/spMmq7j44A?si=ayVjBvoDawel0o3h",
  "IS_PRETEST": 0,
  "IS_DELETED": 0,
  "TIME_LEVEL": "2024-10-09T14:23:46.000Z",
  "ID_STUDENT_LEARNING": null,
  "SCORE": null
},
{
  "NAME_SECTION": "Listening",
  "NAME_TOPIC": "Belajar Memahami Pratista",
  "ID_LEVEL":
"fd783247-3428-4ecc-aelf-227470c0fca2",
  "ID_TOPIC":
"891f6b2e-b385-4dc5-9bbd-97a44daf7138",
  "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
  "NAME_LEVEL": "Level 2",
  "CONTENT": "Ini tes bang",
  "AUDIO":
"audio-fd783247-3428-4ecc-aelf-227470c0fca2-6174deec821539477b2
9c0b1aa6ff375.mp3",
  "IMAGE":
"image-fd783247-3428-4ecc-aelf-227470c0fca2-876b42211a481cce1c3
a36b42979f3a5.png",
  "VIDEO":
"https://youtu.be/spMmq7j44A?si=ayVjBvoDawel0o3h",
  "IS_PRETEST": 0,
  "IS_DELETED": 0,
  "TIME_LEVEL": "2024-08-29T07:54:26.000Z",
  "ID_STUDENT_LEARNING": null,
  "SCORE": null
},
{

```

```

        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Pratista",
        "ID_LEVEL":
"82ccdd34-df40-4dd3-9762-e3cab1e5368d",
        "ID_TOPIC":
"891f6b2e-b385-4dc5-9bbd-97a44daf7138",
        "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
        "NAME_LEVEL": "Level 3",
        "CONTENT": "Ini tes bang",
        "AUDIO":
"AUDIO-b964bce4-f29f-4642-801d-8382b71a8fc9-d9937102a559fbce6d6
c55d66e799066.mp3",
        "IMAGE":
"image-82ccdd34-df40-4dd3-9762-e3cab1e5368d-876b42211a481cce1c3
a36b42979f3a5.png",
        "VIDEO":
"https://youtu.be/spMmq7j44A?si=ayVjBvoDawel0o3h",
        "IS_PRETEST": 0,
        "IS_DELETED": 0,
        "TIME_LEVEL": "2024-08-29T07:55:19.000Z",
        "ID_STUDENT_LEARNING": null,
        "SCORE": null
    },
    {
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Pratista",
        "ID_LEVEL":
"57b57bb5-101e-458a-a433-aa4954b625e6",
        "ID_TOPIC":
"891f6b2e-b385-4dc5-9bbd-97a44daf7138",
        "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
        "NAME_LEVEL": "Level 4",
        "CONTENT": "Ini tes bang",
        "AUDIO":
"audio-57b57bb5-101e-458a-a433-aa4954b625e6-6174deec821539477b2
9c0b1aa6ff375.mp3",
        "IMAGE":
"image-57b57bb5-101e-458a-a433-aa4954b625e6-876b42211a481cce1c3
a36b42979f3a5.png",
        "VIDEO":
"https://youtu.be/07Ll4UH_PgM?si=MHNu67zsc4Z0j3_f",
        "IS_PRETEST": 0,

```

```

        "IS_DELETED": 0,
        "TIME_LEVEL": "2024-08-29T07:55:01.000Z",
        "ID_STUDENT_LEARNING":
        "e7747f39-3d76-4c63-86c9-330904208c6b",
        "SCORE": null
    },
    {
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Pratista",
        "ID_LEVEL":
        "9d8a9687-92ed-4d96-8e03-a0dc0720bdc5",
        "ID_TOPIC":
        "891f6b2e-b385-4dc5-9bbd-97a44daf7138",
        "ID_SECTION":
        "49390272-efbd-4f1c-9b63-39fa8905b891",
        "NAME_LEVEL": "Level 5",
        "CONTENT": "Ini tes bang",
        "AUDIO":
        "audio-9d8a9687-92ed-4d96-8e03-a0dc0720bdc5-6174deec821539477b2
        9c0b1aa6ff375.mp3",
        "IMAGE":
        "image-9d8a9687-92ed-4d96-8e03-a0dc0720bdc5-876b42211a481cce1c3
        a36b42979f3a5.png",
        "VIDEO":
        "https://youtu.be/spMmq7j44A?si=ayVjBvoDawel0o3h",
        "IS_PRETEST": 0,
        "IS_DELETED": 0,
        "TIME_LEVEL": "2024-08-29T07:55:22.000Z",
        "ID_STUDENT_LEARNING": null,
        "SCORE": null
    },
    {
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Pratista",
        "ID_LEVEL":
        "0f89df44-edf8-40f6-a81b-1bc77628d798",
        "ID_TOPIC":
        "891f6b2e-b385-4dc5-9bbd-97a44daf7138",
        "ID_SECTION":
        "49390272-efbd-4f1c-9b63-39fa8905b891",
        "NAME_LEVEL": "Level 6",
        "CONTENT": "Ini tes bang",
        "AUDIO": null,

```

```

"IMAGE":
"image-0f89df44-edf8-40f6-a81b-1bc77628d798-876b42211a481cce1c3
a36b42979f3a5.png",
    "VIDEO": null,
    "IS_PRETEST": 0,
    "IS_DELETED": 0,
    "TIME_LEVEL": "2024-09-04T02:22:43.000Z",
    "ID_STUDENT_LEARNING":
"6b81668b-42e2-4254-af44-74063ac61ece",
    "SCORE": 94
  }
]
}
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
404	Not Found	Topic not found
500	Server Error	Internal Server Error

Tabel 89. Status Code Get Level by Topic Id

d. Get Level For Admin Page

Endpoint ini digunakan untuk mendapatkan seluruh data level yang ada untuk halaman admin dengan beberapa parameter untuk pagination, search, dan sort. Dibawah ini adalah route serta method yang digunakan.

Route	/api/level/admin?page=&limit=&search=&sort=
Method	GET
Request	JSON raw

Tabel 90. Route Get Level For Admin Page

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
page	params	int	No	page yang ingin ditampilkan.
limit	params	int	No	limit dari banyaknya data yang akan ditampilkan pada sebuah page (default adalah 10).
search	params	String	No	parameter untuk melakukan search data level.
sort	params	String	No	parameter untuk melakukan sort.

Tabel 91. *Parameter Get Level For Admin Page*

- Response

```
{
  "payload": {
    "levels": [
      {
        "ID_LEVEL": "d107216f-a8e2-4234-9a1a-dd119029008d",
        "ID_SECTION": "49390272-efbd-4f1c-9b63-39fa8905b891",
        "ID_TOPIC": "c2453b27-c9df-4f24-8502-ae32bd8044ad",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Naresh",
        "NAME_LEVEL": "Pretest",
        "TIME_LEVEL": "2024-09-18T04:01:01.000Z"
      },
      {
        "ID_LEVEL": "37fc6d77-5974-489d-80ac-855f112a61d5",
        "ID_SECTION": "49390272-efbd-4f1c-9b63-39fa8905b891",
        "ID_TOPIC": "c2453b27-c9df-4f24-8502-ae32bd8044ad",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Naresh",
        "NAME_LEVEL": "Level 1",

```

```

        "TIME_LEVEL": "2024-08-27T01:53:27.000Z"
    },
    {
        "ID_LEVEL":
"66b2eea5-fc98-4ddl-b557-49bb6b1d02f8",
        "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
        "ID_TOPIC":
"c2453b27-c9df-4f24-8502-ae32bd8044ad",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Naresh",
        "NAME_LEVEL": "Level 2",
        "TIME_LEVEL": "2024-08-27T01:47:33.000Z"
    },
    {
        "ID_LEVEL":
"41c20abf-fefa-4f2d-a1e9-12a881ecc241",
        "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
        "ID_TOPIC":
"c2453b27-c9df-4f24-8502-ae32bd8044ad",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Naresh",
        "NAME_LEVEL": "Level 3",
        "TIME_LEVEL": "2024-08-27T20:14:32.000Z"
    },
    {
        "ID_LEVEL":
"39417f90-ca22-44d4-93e0-dfe2581488ce",
        "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
        "ID_TOPIC":
"c2453b27-c9df-4f24-8502-ae32bd8044ad",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Naresh",
        "NAME_LEVEL": "Level 4",
        "TIME_LEVEL": "2024-08-27T20:17:51.000Z"
    },
    {
        "ID_LEVEL":
"f4602333-c35c-4847-9ae0-b41972783fae",
        "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",

```

```

"ID_TOPIC":
"c2453b27-c9df-4f24-8502-ae32bd8044ad",
    "NAME_SECTION": "Listening",
    "NAME_TOPIC": "Belajar Memahami Naresh",
    "NAME_LEVEL": "Level 5",
    "TIME_LEVEL": "2024-10-09T08:40:53.000Z"
  },
  {
    "ID_LEVEL":
"cb315d09-230c-4b3b-ad5b-7ea12ef43d4e",
    "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
    "ID_TOPIC":
"c2453b27-c9df-4f24-8502-ae32bd8044ad",
    "NAME_SECTION": "Listening",
    "NAME_TOPIC": "Belajar Memahami Naresh",
    "NAME_LEVEL": "Level 6",
    "TIME_LEVEL": "2024-09-18T03:58:42.000Z"
  }
],
"currentPage": 1,
"totalPages": 2,
"totalItems": 13
},
"statusCode": 200,
"message": "Levels retrieved successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Levels retrieved successfully
500	Server Error	Error retrieving levels data!

Tabel 92. Status Code Get Level For Admin Page

e. Get Level Files

Endpoint ini digunakan untuk mendapatkan data media yang ada pada level id yang diberikan. Dibawah ini adalah route serta method yang digunakan.

Route	/api/level/file/:levelId
-------	--------------------------

Method	GET
Request	JSON raw

Tabel 93. *Route Get Level Files*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
levelId	params	String	Yes	id dari level yang medianya ingin di tampilkan.

Tabel 94. *Parameter Get Level Files*

- Response

```
{
  "payload": {
    "audioFiles": [
      "AUDIO-09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878-29ae9e200ce22524778e52421dc89c8c.mp3",
      "AUDIO-09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878-6174deec821539477b29c0b1aa6ff375.mp3"
    ],
    "imageFiles": [
      "IMAGE-09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878-11f0616a8c96098577643b09f2341f3b.jpeg",
      "IMAGE-09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878-876b42211a481cce1c3a36b42979f3a5.png",
      "IMAGE-09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878-9b7abd31eb795a753c654509ff2357ea.jpeg",
      "IMAGE-09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878-ef42b968ff137775484cd8d9ce1397ef.jpeg"
    ]
  }
}
```

```

},
"statusCode": 200,
"message": "Files retrieved successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Files retrieved successfully
500	Server Error	Internal Server Error

Tabel 95. Status Code Get Level Files

f. Update Level

Endpoint ini digunakan untuk melakukan update data level yang ada pada sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/level/:id
Method	PUT
Request	form-data

Tabel 96. Route Update Level

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	Berisikan id dari level yang ingin diupdate.
NAME_LEVEL	body	String	No	Berisikan nama baru untuk level yang ingin diupdate.
ID_SECTION	body	String	Yes	Berisikan id_section dari yang berhubungan dengan topic dari level ini.

ID_TOPI C	body	String	Yes	Berisikan id_topic yang berkaitan dengan section dan level ini.
CONTEN T	body	Longte xt	No	Berisikan content dalam bentuk html code.
VIDEO	body	String	No	Berisikan link video yang ingin ditambahkan atau diupdate.
AUDIO	body	file	No	Berisikan file audio yang ingin diupdate.
IMAGE	body	file	No	Berisikan file image yang ingin diupdate.

Tabel 97. *Parameter Update Level*

- Response

```
{
  "payload": {
    "ID_LEVEL": "d107216f-a8e2-4234-9a1a-dd119029008d",
    "ID_TOPIC": "c2453b27-c9df-4f24-8502-ae32bd8044ad",
    "ID_SECTION": "49390272-efbd-4f1c-9b63-39fa8905b891",
    "NAME_LEVEL": "Pretest",
    "CONTENT": "Ini tes bang",
    "AUDIO": "AUDIO-d107216f-a8e2-4234-9a1a-dd119029008d-d6dbdf591fc91b865f7797a7b778afc9.mp3",
    "IMAGE": "IMAGE-d107216f-a8e2-4234-9a1a-dd119029008d-e1567771f810befaf146065bf799e38f.jpeg",
    "VIDEO": "https://youtu.be/07Ll4UH_PgM?si=MHNu67zsc4Z0j3_f",
    "IS_PRETEST": 1,
    "ROUTE_1": "37fc6d77-5974-489d-80ac-855f112a61d5",
    "ROUTE_2": "66b2eea5-fc98-4dd1-b557-49bb6b1d02f8",
    "ROUTE_3": "41c20abf-fefa-4f2d-a1e9-12a881ecc241",
    "ROUTE_4": "39417f90-ca22-44d4-93e0-dfe2581488ce",
    "ROUTE_5": "c3a35e3f-32c3-44aa-abd8-f8b5a3fc2131",
    "ROUTE_6": "cb315d09-230c-4b3b-ad5b-7ea12ef43d4e",
    "TIME_LEVEL": "2024-09-18T04:01:01.000Z"
  },
  "statusCode": 200,
  "message": "Level updated successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Level updated successfully
400	Bad Request	Topic does not relate to the provided Section
404	Not Found	Level not found
409	Conflict	A level with this name already exists under this topic
500	Error Server	Internal Server Error.

Tabel 98. *Status Code Update Level*

g. Upload Level File

Endpoint ini digunakan untuk melakukan upload file ke sistem, file bisa berupa audio ataupun image. Dibawah ini adalah route serta method yang digunakan.

Route	/api/level/file/:levelId
Method	POST
Request	form-data

Tabel 99. *Route Upload Level File*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
levelId	params	String	Yes	Berisikan id level yang ingin ditambahkan media terkait.

AUDIO[i]	body	file	No	Berisikan file audio yang ingin diinputkan ke sistem (bisa banyak file dengan index) dengan batas upload adalah 10MB.
IMAGE[i]	body	file	No	Berisikan file image yang ingin diinputkan ke sistem (bisa banyak file dengan index) dengan batas upload adalah 10MB.

Tabel 100. *Parameter Upload Level File*

- Response

```
{
  "payload": {
    "IMAGE[0]":
"IMAGE-09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878-11f0616a8c960985776
43b09f2341f3b.jpeg",
    "IMAGE[1]":
"IMAGE-09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878-876b42211a481cce1c3
a36b42979f3a5.png",
    "AUDIO[0]":
"AUDIO-09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878-29ae9e200ce22524778
e52421dc89c8c.mp3",
    "AUDIO[1]":
"AUDIO-09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878-6174deec821539477b2
9c0b1aa6ff375.mp3"
  },
  "statusCode": 200,
  "message": "Files uploaded successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Files uploaded successfully
400	Bad Request	No files to upload

404	Not Found	Level not found
500	Error Server	Internal Server Error.

Tabel 101. *Status Code Upload Level File*

h. Delete Level File by Id and Levelname

Endpoint ini digunakan untuk melakukan penghapusan file yang telah diupload untuk sebuah level/material. Dibawah ini adalah route serta method yang digunakan.

Route	/api/level/file/:id
Method	DELETE
Request	JSON raw

Tabel 102. *Route Delete Level File by Id and Levelname*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari level yang medianya ingin dihapus
filename	body	String	Yes	nama dari file yang ingin dihapus

Tabel 103. *Parameter Delete Level File by Id and Levelname*

- Response

```
{
  "payload": null,
  "statusCode": 200,
  "message": "AUDIO file deleted successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	{fileType} file deleted successfully
400	Bad Request	Level ID in file name does not match
404	Not Found	Level not found
500	Error Server	Internal Server Error.

Tabel 104. *Status Code Delete Level File by Id and Levelname*

i. Delete Level

Endpoint ini digunakan untuk melakukan penghapusan data level dari sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/level/:id
Method	DELETE
Request	JSON raw

Tabel 105. *Route Delete Level*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari level yang ingin dihapus

Tabel 106. *Parameter Delete Level*

- Response

```
{
  "payload": null,
  "statusCode": 200,
  "message": "Level and related exercises soft deleted successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Level and related exercises soft deleted successfully
404	Not Found	Level not found
500	Error Server	Internal Server Error.

Tabel 107. *Status Code Delete Level*

6. Exercise Model

Pada Exercise Model, berisi endpoint yang berkaitan dengan manipulasi data exercise dalam sistem. Koleksi API ini mencakup fungsi seperti *Get Exercises*, *Get Exercise by Id*, *Get Exercise By Level Id*, *Get Exercise By Level Id for Admin*, *Get Exercise For Admin Page*, *Create Exercise*, *Update Exercise*, *Update Exercise Title*, *Delete Exercise File* dan *Delete Exercise*.

a. Get Exercises

Endpoint ini digunakan untuk mendapatkan seluruh data exercise yang ada. Dibawah ini adalah route serta method yang digunakan.

Route	/api/exercise
Method	GET
Request	JSON raw

Tabel 108. *Route Get Exercises*

- Parameters

Pada endpoint ini, tidak perlu mengirimkan parameter apapun, cukup memanggil route beserta methodnya yang nantinya akan divalidasi oleh sistem backend.

- Response

```
{
  "payload": [
    {
```

```

      "ID_ADMIN_EXERCISE": "009f53d1-d1e0-4cdd-b857-14d1e7a88efb",
      "ID_LEVEL": "17bad807-45c3-44e6-b1b5-8bd52fa22079",
      "TITLE": "Soal 4",
      "QUESTION": "Where does Naresh's journey begin?",
      "SCORE_WEIGHT": 5,
      "QUESTION_TYPE": "MCQ",
      "AUDIO":
      "AUDIO-009f53d1-d1e0-4cdd-b857-14d1e7a88efb-453784d90757a3577d6
      ebe66791a41f3.mp3",
      "VIDEO": null,
      "IMAGE": null,
      "IS_DELETED": 0,
      "TIME_ADMIN_EXC": "2024-09-19T08:40:45.000Z",
      "multipleChoices": [
        {
          "ID_MULTIPLE_CHOICES": "ec429417-4106-470c-af61-a3aa98ec51b4",
          "ID_ADMIN_EXERCISE": "009f53d1-d1e0-4cdd-b857-14d1e7a88efb",
          "OPTION_A": "The Western Desert",
          "OPTION_B": "The Sacred Forest",
          "OPTION_C": "The Mountain of Dreams",
          "OPTION_D": "The Village of Dawn",
          "OPTION_E": "The River of Souls",
          "ANSWER_KEY": "D",
          "TIME_MULTIPLE_CHOICES": "2024-09-19T08:40:45.000Z"
        }
      ]
    },
    {
      "ID_ADMIN_EXERCISE": "0a38f532-c028-4034-8c2c-737ef9be5a6d",
      "ID_LEVEL": "17bad807-45c3-44e6-b1b5-8bd52fa22079",
      "TITLE": "Soal 3",
      "QUESTION": "Match the supporting characters to
their roles in the story:",
      "SCORE_WEIGHT": 10,
      "QUESTION_TYPE": "MPQ",
      "AUDIO": null,
      "VIDEO": null,
      "IMAGE":
      "IMAGE-0a38f532-c028-4034-8c2c-737ef9be5a6d-58e4b8fcbf104d7b100
      c4a3c12abf306.jpeg",
      "IS_DELETED": 0,
      "TIME_ADMIN_EXC": "2024-09-19T08:38:26.000Z",
      "matchingPairs": [
        {

```

```

    "ID_MATCHING_PAIRS": "1707ebb2-aaf5-4091-bae4-d47ce6a4b939",
    "ID_ADMIN_EXERCISE": "0a38f532-c028-4034-8c2c-737ef9be5a6d",
    "LEFT_PAIR": "Village Elder",
    "RIGHT_PAIR": "Warns Naresh of dangers",
    "TIME_MATCHING_PAIRS": "2024-09-19T08:38:26.000Z"
  },
  {
    "ID_MATCHING_PAIRS": "6550798c-85b5-4b42-b23d-7b282321b8b0",
    "ID_ADMIN_EXERCISE": "0a38f532-c028-4034-8c2c-737ef9be5a6d",
    "LEFT_PAIR": "Merchant",
    "RIGHT_PAIR": "Sells Naresh important supplies",
    "TIME_MATCHING_PAIRS": "2024-09-19T08:38:26.000Z"
  },
  {
    "ID_MATCHING_PAIRS": "a634341b-6809-45d9-85b9-50b1b99b8bbd",
    "ID_ADMIN_EXERCISE": "0a38f532-c028-4034-8c2c-737ef9be5a6d",
    "LEFT_PAIR": "Animal Guide",
    "RIGHT_PAIR": "Guides Naresh with wisdom",
    "TIME_MATCHING_PAIRS": "2024-09-19T08:38:26.000Z"
  },
  {
    "ID_MATCHING_PAIRS": "c3c4fe3b-cfe8-4d9a-81d6-b140ee653c5a",
    "ID_ADMIN_EXERCISE": "0a38f532-c028-4034-8c2c-737ef9be5a6d",
    "LEFT_PAIR": "Blacksmith",
    "RIGHT_PAIR": "Provides a key to Naresh's
journey",
    "TIME_MATCHING_PAIRS": "2024-09-19T08:38:26.000Z"
  },
  {
    "ID_MATCHING_PAIRS": "e021da95-99f6-4a64-9104-f8e5a6aa9737",
    "ID_ADMIN_EXERCISE": "0a38f532-c028-4034-8c2c-737ef9be5a6d",
    "LEFT_PAIR": "Sorceress",
    "RIGHT_PAIR": "Teaches Naresh magical skills",
    "TIME_MATCHING_PAIRS": "2024-09-19T08:38:26.000Z"
  }
]
},
],
"statusCode": 200,
"message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Internal Server Error

Tabel 109. *Status Code Get Exercises*

b. Get Exercise by Id

Endpoint ini digunakan untuk mendapatkan data exercise yang ada berdasarkan id. Dibawah ini adalah route serta method yang digunakan.

Route	/api/exercise/:id
Method	GET
Request	JSON raw

Tabel 110. *Route Get Exercise by Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari exercise yang ingin di tampilkan.

Tabel 111. *Parameter Get Exercise by Id*

- Response

```
{
  "payload": {
    "ID_ADMIN_EXERCISE": "42cbd79f-ad94-404c-9c9d-51b77b414ade",
    "ID_LEVEL": "57b57bb5-101e-458a-a433-aa4954b625e6",
    "TITLE": "Soal 5",
    "QUESTION": "Apakah Naresh itu pratista?",
    "SCORE_WEIGHT": 2,
    "QUESTION_TYPE": "TFQ",
    "AUDIO": null,
    "VIDEO": "https://youtu.be/07Ll4UH_PgM?si=MHNu67zsc4Z0j3_f",
    "IMAGE": null,
    "IS_DELETED": 0,
    "TIME_ADMIN_EXC": "2024-09-09T05:51:20.000Z",
```

```

    "trueFalse": [
      {
        "ID_TRUE_FALSE": "cb684b51-a525-49ba-a0e3-4cd2fd60c516",
        "ID_ADMIN_EXERCISE": "42cbd79f-ad94-404c-9c9d-51b77b414ade",
        "TIME_TRUE_FALSE": "2024-09-09T05:51:20.000Z"
      }
    ],
    "statusCode": 200,
    "message": "Success"
  }
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
404	Not Found	Exercise not found
500	Server Error	Internal Server Error

Tabel 112. *Status Code Get Exercise by Id*

c. Get Exercise by Level Id

Endpoint ini digunakan untuk mendapatkan data exercise yang ada berdasarkan level id yang terkait. Dibawah ini adalah route serta method yang digunakan.

Route	/api/exercise/level/:idLevel
Method	GET
Request	JSON raw

Tabel 113. *Route Get Exercise by Level Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
-----------	------	-----------	----------	-------------

idLevel	params	String	Yes	id dari level yang exercisenya ditampilkan
---------	--------	--------	-----	--

Tabel 114. *Parameter Get Exercise by Level Id*

- Response

```
{
  "payload": {
    "NAME_SECTION": "Listening",
    "NAME_TOPIC": "Belajar Memahami Pratista",
    "NAME_LEVEL": "Level 6",
    "EXERCISES": [
      {
        "ID_ADMIN_EXERCISE":
"15b4d936-24a7-4842-af1f-6e8ac9f35ab9",
        "ID_LEVEL":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
        "TITLE": "Soal 1",
        "QUESTION": "Apakah Naresh itu pratista?",
        "SCORE_WEIGHT": 2,
        "QUESTION_TYPE": "MCQ",
        "AUDIO":
"audio-15b4d936-24a7-4842-af1f-6e8ac9f35ab9-1f3433d8d7ba85e911f62a93bb002b5a.mp3",
        "VIDEO": null,
        "IMAGE": null,
        "IS_DELETED": 0,
        "TIME_ADMIN_EXC": "2024-09-11T06:40:06.000Z",
        "multipleChoices": [
          {
            "ID_MULTIPLE_CHOICES":
"b29cedb5-1fe8-49c8-845f-dcc42b8f6752",
            "ID_ADMIN_EXERCISE":
"15b4d936-24a7-4842-af1f-6e8ac9f35ab9",
            "OPTION_A": "bukan",
            "OPTION_B": "sangat bukan",
            "OPTION_C": "no way",
            "OPTION_D": "naresh adalah senggani",
            "OPTION_E": "senggani adalah naresh",
            "TIME_MULTIPLE_CHOICES":
"2024-09-11T06:40:06.000Z"
          }
        ]
      }
    ]
  }
}
```



```

    },
    {
        "ID_ADMIN_EXERCISE":
"1643d97d-1095-45ab-91a8-38748431fb0b",
        "ID_LEVEL":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
        "TITLE": "Soal 2",
        "QUESTION": "Apakah Naresh itu pratista?",
        "SCORE_WEIGHT": 2,
        "QUESTION_TYPE": "TFQ",
        "AUDIO": null,
        "VIDEO":
"https://youtu.be/07Ll4UH_PgM?si=MHNu67zsc4Z0j3_f",
        "IMAGE": null,
        "IS_DELETED": 0,
        "TIME_ADMIN_EXC": "2024-09-11T06:45:30.000Z",
        "trueFalse": [
            {
                "ID_TRUE_FALSE":
"7fa884dd-2bf1-4e51-8767-bfb2c2d2176d",
                "ID_ADMIN_EXERCISE":
"1643d97d-1095-45ab-91a8-38748431fb0b",
                "TIME_TRUE_FALSE":
"2024-09-11T06:45:30.000Z"
            }
        ]
    },
    {
        "ID_ADMIN_EXERCISE":
"4019e25d-e187-4946-b50a-32258b9f5c3b",
        "ID_LEVEL":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
        "TITLE": "Soal 4",
        "QUESTION": "Apakah Pratista itu Naresh?",
        "SCORE_WEIGHT": 2,
        "QUESTION_TYPE": "MCQ",
        "AUDIO": null,
        "VIDEO":
"https://youtu.be/07Ll4UH_PgM?si=MHNu67zsc4Z0j3_f",
        "IMAGE": null,
        "IS_DELETED": 0,
        "TIME_ADMIN_EXC": "2024-09-11T06:49:22.000Z",
        "multipleChoices": [
            {

```

```

                                "ID_MULTIPLE_CHOICES":
"4629ccae-2eef-4a3e-9c9d-f8b03566486e",
                                "ID_ADMIN_EXERCISE":
"4019e25d-e187-4946-b50a-32258b9f5c3b",
                                "OPTION_A": "bukan",
                                "OPTION_B": "sangat bukan",
                                "OPTION_C": "no way",
                                "OPTION_D": "naresh adalah senggani",
                                "OPTION_E": "senggani adalah naresh",
                                "TIME_MULTIPLE_CHOICES":
"2024-09-11T06:49:22.000Z"
                                }
                                ],
                                },
                                {
                                "ID_ADMIN_EXERCISE":
"58f59acf-5940-4912-bc7c-2cf7b18ac17b",
                                "ID_LEVEL":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
                                "TITLE": "Soal 6",
                                "QUESTION": "Apakah Pratista itu Senggani?",
                                "SCORE_WEIGHT": 2,
                                "QUESTION_TYPE": "TFQ",
                                "AUDIO": null,
                                "VIDEO": null,
                                "IMAGE": null,
                                "IS_DELETED": 0,
                                "TIME_ADMIN_EXC": "2024-09-11T06:50:08.000Z",
                                "trueFalse": [
                                {
                                "ID_TRUE_FALSE":
"e071e762-2340-4914-b5b7-8f584370fa51",
                                "ID_ADMIN_EXERCISE":
"58f59acf-5940-4912-bc7c-2cf7b18ac17b",
                                "TIME_TRUE_FALSE":
"2024-09-11T06:50:08.000Z"
                                }
                                ],
                                },
                                {
                                "ID_ADMIN_EXERCISE":
"5a52b52f-ad04-4652-b9c3-6f9c213111c9",
                                "ID_LEVEL":
"0f89df44-edf8-40f6-a81b-1bc77628d798",

```

```

        "TITLE": "Soal 3",
        "QUESTION": "Siapa saja yang termasuk pada
semanggi man?",
        "SCORE_WEIGHT": 2,
        "QUESTION_TYPE": "MPQ",
        "AUDIO": null,
        "VIDEO":
"https://youtu.be/07Ll4UH_PgM?si=MHNu67zsc4Z0j3_f",
        "IMAGE": null,
        "IS_DELETED": 0,
        "TIME_ADMIN_EXC": "2024-09-11T06:46:12.000Z",
        "matchingPairs": [
            {
                "ID_MATCHING_PAIRS":
"50e85771-3026-4564-a2bf-9fc3a800192e",
                "ID_ADMIN_EXERCISE":
"5a52b52f-ad04-4652-b9c3-6f9c213111c9",
                "LEFT_PAIR": "Naresh",
                "RIGHT_PAIR": "Senggani",
                "TIME_MATCHING_PAIRS":
"2024-09-11T06:46:13.000Z"
            },
            {
                "ID_MATCHING_PAIRS":
"ec51516d-7372-4630-9dea-7ff2db7b0f61",
                "ID_ADMIN_EXERCISE":
"5a52b52f-ad04-4652-b9c3-6f9c213111c9",
                "LEFT_PAIR": "Senggani",
                "RIGHT_PAIR": "Naresh",
                "TIME_MATCHING_PAIRS":
"2024-09-11T06:46:13.000Z"
            },
            {
                "ID_MATCHING_PAIRS":
"f20246a9-af82-491b-892a-28687a804258",
                "ID_ADMIN_EXERCISE":
"5a52b52f-ad04-4652-b9c3-6f9c213111c9",
                "LEFT_PAIR": "Senggani",
                "RIGHT_PAIR": "Pratista",
                "TIME_MATCHING_PAIRS":
"2024-09-11T06:46:13.000Z"
            }
        ]
    },

```

```

{
    "ID_ADMIN_EXERCISE":
"7502cdfd-4237-4307-941a-63e0403b6c31",
    "ID_LEVEL":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
    "TITLE": "Soal 5",
    "QUESTION": "Apakah Naresh itu Senggani?",
    "SCORE_WEIGHT": 2,
    "QUESTION_TYPE": "TFQ",
    "AUDIO":
"audio-7502cdfd-4237-4307-941a-63e0403b6c31-1f3433d8d7ba85e911f
62a93bb002b5a.mp3",
    "VIDEO": null,
    "IMAGE": null,
    "IS_DELETED": 0,
    "TIME_ADMIN_EXC": "2024-09-11T06:49:54.000Z",
    "trueFalse": [
        {
            "ID_TRUE_FALSE":
"481ee324-36f6-41ac-9f53-770f0b998a67",
            "ID_ADMIN_EXERCISE":
"7502cdfd-4237-4307-941a-63e0403b6c31",
            "TIME_TRUE_FALSE":
"2024-09-11T06:49:54.000Z"
        }
    ]
}
},
"statusCode": 200,
"message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
404	Not Found	Level not found
500	Server Error	Internal Server Error

Tabel 115. *Status Code Get Exercise by Level Id*

d. Get Exercise by Level Id for Admin

Endpoint ini digunakan untuk mendapatkan data exercise yang ada berdasarkan level id yang terkait untuk halaman admin. Dibawah ini adalah route serta method yang digunakan.

Route	/api/exercise/admin/level/:idLevel
Method	GET
Request	JSON raw

Tabel 116. Route Get Exercise by Level Id for Admin

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
idLevel	params	String	Yes	id dari level yang exercisenya ditampilkan

Tabel 117. Parameter Get Exercise by Level Id for Admin

- Response

```
{
  "payload": {
    "NAME_SECTION": "Listening",
    "NAME_TOPIC": "Belajar Memahami Pratista",
    "NAME_LEVEL": "Level 6",
    "EXERCISES": [
      {
        "ID_ADMIN_EXERCISE":
"15b4d936-24a7-4842-af1f-6e8ac9f35ab9",
        "ID_LEVEL":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
        "TITLE": "Soal 1",
        "QUESTION": "Apakah Naresh itu pratista?",
        "SCORE_WEIGHT": 2,
        "QUESTION_TYPE": "MCQ",
```

```

"audio-15b4d936-24a7-4842-af1f-6e8ac9f35ab9-1f3433d8d7ba85e911f62a93bb002b5a.mp3",
    "VIDEO": null,
    "IMAGE": null,
    "IS_DELETED": 0,
    "TIME_ADMIN_EXC": "2024-09-11T06:40:06.000Z",
    "multipleChoices": [
        {
            "ID_MULTIPLE_CHOICES":
"b29cedb5-1fe8-49c8-845f-dcc42b8f6752",
            "ID_ADMIN_EXERCISE":
"15b4d936-24a7-4842-af1f-6e8ac9f35ab9",
            "OPTION_A": "bukan",
            "OPTION_B": "sangat bukan",
            "OPTION_C": "no way",
            "OPTION_D": "naresh adalah senggani",
            "OPTION_E": "senggani adalah naresh",
            "ANSWER_KEY": "D",
            "TIME_MULTIPLE_CHOICES":
"2024-09-11T06:40:06.000Z"
        }
    ],
    {
        "ID_ADMIN_EXERCISE":
"1643d97d-1095-45ab-91a8-38748431fb0b",
        "ID_LEVEL":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
        "TITLE": "Soal 2",
        "QUESTION": "Apakah Naresh itu pratista?",
        "SCORE_WEIGHT": 2,
        "QUESTION_TYPE": "TFQ",
        "AUDIO": null,
        "VIDEO":
"https://youtu.be/07Ll4UH_PgM?si=MHNu67zsc4Z0j3_f",
        "IMAGE": null,
        "IS_DELETED": 0,
        "TIME_ADMIN_EXC": "2024-09-11T06:45:30.000Z",
        "trueFalse": [
            {
                "ID_TRUE_FALSE":
"7fa884dd-2bf1-4e51-8767-bfb2c2d2176d",

```

```

"ID_ADMIN_EXERCISE":
"1643d97d-1095-45ab-91a8-38748431fb0b",
    "IS_TRUE": 0,
    "TIME_TRUE_FALSE":
"2024-09-11T06:45:30.000Z"
    }
    ],
    },
    {
        "ID_ADMIN_EXERCISE":
"5a52b52f-ad04-4652-b9c3-6f9c213111c9",
        "ID_LEVEL":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
        "TITLE": "Soal 3",
        "QUESTION": "Siapa saja yang termasuk pada
semanggi man?",
        "SCORE_WEIGHT": 2,
        "QUESTION_TYPE": "MPQ",
        "AUDIO": null,
        "VIDEO":
"https://youtu.be/07Ll4UH_PgM?si=MHNu67zsc4Z0j3_f",
        "IMAGE": null,
        "IS_DELETED": 0,
        "TIME_ADMIN_EXC": "2024-09-11T06:46:12.000Z",
        "matchingPairs": [
            {
                "ID_MATCHING_PAIRS":
"50e85771-3026-4564-a2bf-9fc3a800192e",
                "ID_ADMIN_EXERCISE":
"5a52b52f-ad04-4652-b9c3-6f9c213111c9",
                "LEFT_PAIR": "Naresh",
                "RIGHT_PAIR": "Senggani",
                "TIME_MATCHING_PAIRS":
"2024-09-11T06:46:13.000Z"
            },
            {
                "ID_MATCHING_PAIRS":
"ec51516d-7372-4630-9dea-7ff2db7b0f61",
                "ID_ADMIN_EXERCISE":
"5a52b52f-ad04-4652-b9c3-6f9c213111c9",
                "LEFT_PAIR": "Senggani",
                "RIGHT_PAIR": "Naresh",
                "TIME_MATCHING_PAIRS":
"2024-09-11T06:46:13.000Z"
            }
        ]
    }
]

```

```

        },
        {
            "ID_MATCHING_PAIRS":
"f20246a9-af82-491b-892a-28687a804258",
            "ID_ADMIN_EXERCISE":
"5a52b52f-ad04-4652-b9c3-6f9c213111c9",
            "LEFT_PAIR": "Senggani",
            "RIGHT_PAIR": "Pratista",
            "TIME_MATCHING_PAIRS":
"2024-09-11T06:46:13.000Z"
        }
    ]
},
{
            "ID_ADMIN_EXERCISE":
"4019e25d-e187-4946-b50a-32258b9f5c3b",
            "ID_LEVEL":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
            "TITLE": "Soal 4",
            "QUESTION": "Apakah Pratista itu Naresh?",
            "SCORE_WEIGHT": 2,
            "QUESTION_TYPE": "MCQ",
            "AUDIO": null,
            "VIDEO":
"https://youtu.be/07Ll4UH_PgM?si=MHNu67zsc4Z0j3_f",
            "IMAGE": null,
            "IS_DELETED": 0,
            "TIME_ADMIN_EXC": "2024-09-11T06:49:22.000Z",
            "multipleChoices": [
                {
                    "ID_MULTIPLE_CHOICES":
"4629ccae-2eef-4a3e-9c9d-f8b03566486e",
                    "ID_ADMIN_EXERCISE":
"4019e25d-e187-4946-b50a-32258b9f5c3b",
                    "OPTION_A": "bukan",
                    "OPTION_B": "sangat bukan",
                    "OPTION_C": "no way",
                    "OPTION_D": "naresh adalah senggani",
                    "OPTION_E": "senggani adalah naresh",
                    "ANSWER_KEY": "E",
                    "TIME_MULTIPLE_CHOICES":
"2024-09-11T06:49:22.000Z"
                }
            ]
}
]

```



```

    },
    {
        "ID_ADMIN_EXERCISE":
"7502cdfd-4237-4307-941a-63e0403b6c31",
        "ID_LEVEL":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
        "TITLE": "Soal 5",
        "QUESTION": "Apakah Naresh itu Senggani?",
        "SCORE_WEIGHT": 2,
        "QUESTION_TYPE": "TFQ",
        "AUDIO":
"audio-7502cdfd-4237-4307-941a-63e0403b6c31-1f3433d8d7ba85e911f
62a93bb002b5a.mp3",
        "VIDEO": null,
        "IMAGE": null,
        "IS_DELETED": 0,
        "TIME_ADMIN_EXC": "2024-09-11T06:49:54.000Z",
        "trueFalse": [
            {
                "ID_TRUE_FALSE":
"481ee324-36f6-41ac-9f53-770f0b998a67",
                "ID_ADMIN_EXERCISE":
"7502cdfd-4237-4307-941a-63e0403b6c31",
                "IS_TRUE": 1,
                "TIME_TRUE_FALSE":
"2024-09-11T06:49:54.000Z"
            }
        ]
    },
    {
        "ID_ADMIN_EXERCISE":
"58f59acf-5940-4912-bc7c-2cf7b18ac17b",
        "ID_LEVEL":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
        "TITLE": "Soal 6",
        "QUESTION": "Apakah Pratista itu Senggani?",
        "SCORE_WEIGHT": 2,
        "QUESTION_TYPE": "TFQ",
        "AUDIO": null,
        "VIDEO": null,
        "IMAGE": null,
        "IS_DELETED": 0,
        "TIME_ADMIN_EXC": "2024-09-11T06:50:08.000Z",
        "trueFalse": [

```

```

{
  "ID_TRUE_FALSE":
  "e071e762-2340-4914-b5b7-8f584370fa51",
  "ID_ADMIN_EXERCISE":
  "58f59acf-5940-4912-bc7c-2cf7b18ac17b",
  "IS_TRUE": 1,
  "TIME_TRUE_FALSE":
  "2024-09-11T06:50:08.000Z"
}
]
}
],
},
"statusCode": 200,
"message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
404	Not Found	Level not found
500	Server Error	Internal Server Error

Tabel 118. Status Code Get Exercise by Level Id for Admin

e. Get Exercise for Admin Page

Endpoint ini digunakan untuk mendapatkan data exercise yang ada pada sistem dengan mengimplementasikan fungsi pagination, search, dan sort. Dibawah ini adalah route serta method yang digunakan.

Route	/api/exercise/admin?page=&limit=&sort=&search=
Method	GET
Request	JSON raw

Tabel 119. Route Get Exercise for Admin Page

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
page	params	int	No	Berisikan page yang ingin ditampilkan dengan nilai default adalah 1
limit	params	int	No	Berisikan limit data yang ditampilkan pada sebuah page dengan nilai default adalah 10
search	params	String	No	Parameter untuk melakukan search data yang ada pada page
sort	params	String	No	Parameter untuk melakukan sorting berdasarkan header yang ada di website.

Tabel 120. *Parameter Get Exercise for Admin Page*

- Response

```
{
  "payload": {
    "exercises": [
      {
        "ID_ADMIN_EXERCISE":
"42cbd79f-ad94-404c-9c9d-51b77b414ade",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Pratista",
        "NAME_LEVEL": "Level 4",
        "TITLE": "Soal 5",
        "QUESTION": "Apakah Naresh itu pratista?",
        "QUESTION_TYPE": "TFQ",
        "ANSWER_KEY": "false",
        "TIME_ADMIN_EXC": "12:51 09/09/2024"
      },
      {
        "ID_ADMIN_EXERCISE":
"6275319a-1fe6-4551-8045-72a48883ae2c",
        "NAME_SECTION": "Listening",
```

```

        "NAME_TOPIC": "Belajar Memahami Pratista",
        "NAME_LEVEL": "Level 4",
        "TITLE": "Soal 1",
        "QUESTION": "Apakah Naresh itu pratista?",
        "QUESTION_TYPE": "MCQ",
        "ANSWER_KEY": "D",
        "TIME_ADMIN_EXC": "09:51 05/09/2024"
    },
    {
        "ID_ADMIN_EXERCISE":
"0ac4b6b3-21d6-4000-88d7-8d9b432f9668",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Pratista",
        "NAME_LEVEL": "Level 4",
        "TITLE": "Soal 3",
        "QUESTION": "Apakah Naresh itu senggani?",
        "QUESTION_TYPE": "TFQ",
        "ANSWER_KEY": "true",
        "TIME_ADMIN_EXC": "15:58 05/09/2024"
    },
    {
        "ID_ADMIN_EXERCISE":
"9a435e16-28ea-4d29-93f3-e33e9a580638",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Pratista",
        "NAME_LEVEL": "Level 4",
        "TITLE": "Soal 2",
        "QUESTION": "Apakah Pratista itu naresh?",
        "QUESTION_TYPE": "MCQ",
        "ANSWER_KEY": "E",
        "TIME_ADMIN_EXC": "10:43 05/09/2024"
    },
    {
        "ID_ADMIN_EXERCISE":
"5320711a-9c0f-4545-a2b9-7ddf3a605947",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Pratista",
        "NAME_LEVEL": "Level 4",
        "TITLE": "Soal 4",
        "QUESTION": "Siapa saja yang termasuk pada
semanggi man?",
        "QUESTION_TYPE": "MPQ",
        "ANSWER_KEY": "Senggani-Pratista,
Naresh-Senggani, Pratista-Senggani",

```

```

        "TIME_ADMIN_EXC": "09:47 09/09/2024"
    },
    ],
    "currentPage": 1,
    "totalPages": 1,
    "totalExercises": 5
},
"statusCode": 200,
"message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Error retrieving exercises data!

Tabel 121. *Status Code Get Exercise for Admin Page*

f. Create Exercise

Endpoint ini digunakan untuk membuat sebuah exercise pada sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/exercises
Method	POST
Request	form-data

Tabel 122. *Route Create Exercise*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
exercises[i][ID_LEVEL]	body	String	Yes	Berisikan id level yang ingin ditambahkan datanya

exercises[i][QUESTION]	body	String	Yes	Berisikan pertanyaan dari exercise tersebut
exercises[i][SCORE_WEIGHT]	body	String	Yes	Berisikan bobot nilai untuk exercise ini.
exercises[i][QUESTION_TYPE]	body	String	Yes	Berisikan tipe soal untuk exercise
exercises[i][VIDEO]	body	String	No	Berisikan link video untuk exercise
AUDIO[i]	body	file	No	Berisikan file audio untuk exercise
IMAGE[i]	body	file	No	Berisikan file image untuk exercise
MCQ (Multiple Choice Question)				
exercises[i][OPTION_A]	body	String	Yes	Berisikan pilihan pertama
exercises[i][OPTION_B]	body	String	Yes	Berisikan pilihan kedua
exercises[i][OPTION_C]	body	String	Yes	Berisikan pilihan ketiga
exercises[i][OPTION_D]	body	String	Yes	Berisikan pilihan keempat
exercises[i][OPTION_E]	body	String	Yes	Berisikan pilihan kelima
exercises[i][ANSWER_KEY]	body	String	Yes	Jawaban dari pilihan yang telah ditambahkan

MPQ (Matching Pairs Question)				
exercises[i][PAIRS][i][LEFT_PAIR]	body	Text	Yes	Pasangan kiri yang akan diinputkan ke exercise
exercises[i][PAIRS][i][RIGHT_PAIR]	body	Text	Yes	Pasangan kanan yang akan diinputkan ke exercise
TFQ (True False Question)				
exercises[i][IS_TRUE]	body	String	Yes	Berisikan jawaban benar atau salah

Tabel 123. *Parameter Create Exercise*

- Response

```
{
  "payload": [
    {
      "ID_ADMIN_EXERCISE":
"6b56e2c2-069a-4c5d-8045-85a52f238acb",
      "IS_DELETED": 0,
      "TIME_ADMIN_EXC": "2024-10-09T05:39:14.441Z",
      "ID_LEVEL": "41c20abf-fefa-4f2d-a1e9-12a881ecc241",
      "TITLE": "Soal 1",
      "QUESTION": "Testing exercises berindex 1",
      "SCORE_WEIGHT": "3",
      "QUESTION_TYPE": "MCQ",
      "AUDIO":
"AUDIO-6b56e2c2-069a-4c5d-8045-85a52f238acb-c4775f42af4a2e1e3e4
d5cb875e261dd.mp3",
      "VIDEO":
"https://youtu.be/spMmq7j44A?si=ayVjBvoDawel0o3h",
      "IMAGE":
"IMAGE-6b56e2c2-069a-4c5d-8045-85a52f238acb-df84fe4f264501985fe
2845d0e43a19e.jpeg",
      "questionDetails": {
        "ID_MULTIPLE_CHOICES":
"30f34be8-d4f3-41b5-a30d-004eee18e36d",
        "TIME_MULTIPLE_CHOICES":
"2024-10-09T05:39:14.456Z",
```

```

"ID_ADMIN_EXERCISE":
"6b56e2c2-069a-4c5d-8045-85a52f238acb",
    "OPTION_A": "apa",
    "OPTION_B": "itu",
    "OPTION_C": "multiple",
    "OPTION_D": "choice",
    "OPTION_E": "question",
    "ANSWER_KEY": "E"
  }
},
{
    "ID_ADMIN_EXERCISE":
"69312f33-7d65-40c5-9600-8fc6d197a6ae",
    "IS_DELETED": 0,
    "TIME_ADMIN_EXC": "2024-10-09T05:39:14.468Z",
    "ID_LEVEL": "41c20abf-fefa-4f2d-a1e9-12a881ecc241",
    "TITLE": "Soal 2",
    "QUESTION": "Testing exercise berindex 2",
    "SCORE_WEIGHT": "5",
    "QUESTION_TYPE": "MPQ",
    "AUDIO":
"AUDIO-69312f33-7d65-40c5-9600-8fc6d197a6ae-c4775f42af4a2e1e3e4
d5cb875e261dd.mp3",
    "VIDEO":
"https://youtu.be/spMmzq7j44A?si=ayVjBvoDawel0o3h",
    "IMAGE": null,
    "questionDetails": [
      {
        "LEFT_PAIR": "ini",
        "RIGHT_PAIR": "pasangan 1"
      },
      {
        "LEFT_PAIR": "pasangan 2",
        "RIGHT_PAIR": "ini"
      }
    ]
  },
  {
    "ID_ADMIN_EXERCISE":
"171da194-1048-45f8-8561-ae000621f3ed",
    "IS_DELETED": 0,
    "TIME_ADMIN_EXC": "2024-10-09T05:39:14.483Z",
    "ID_LEVEL": "41c20abf-fefa-4f2d-a1e9-12a881ecc241",
    "TITLE": "Soal 3",

```



```

        "QUESTION": "Testing exercise berindex 3",
        "SCORE_WEIGHT": "3",
        "QUESTION_TYPE": "TFQ",
        "AUDIO": "AUDIO-171da194-1048-45f8-8561-ae000621f3ed-c4775f42af4a2e1e3e4d5cb875e261dd.mp3",
        "VIDEO": "https://youtu.be/spMmzq7j44A?si=ayVjBvoDawel10o3h",
        "IMAGE": "IMAGE-171da194-1048-45f8-8561-ae000621f3ed-df84fe4f264501985fe2845d0e43a19e.jpeg",
        "questionDetails": {
            "ID_TRUE_FALSE": "cea8a5c9-e2d5-48d9-bbc6-824aa2a8ca8d",
            "TIME_TRUE_FALSE": "2024-10-09T05:39:14.489Z",
            "ID_ADMIN_EXERCISE": "171da194-1048-45f8-8561-ae000621f3ed",
            "IS_TRUE": "1"
        }
    },
    "statusCode": 201,
    "message": "Exercises created successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
201	Created	Exercises created successfully
400	Bad Request	Exercises array is required
500	Server Error	Internal Server Error

Tabel 124. *Status Code Create Exercise*

g. Update Exercise

Endpoint ini digunakan untuk melakukan update sebuah exercise pada sistem.

Dibawah ini adalah route serta method yang digunakan.

Route	/api/exercise/:id
-------	-------------------

Method	PUT
Request	form-data

Tabel 125. *Route Update Exercise*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari exercise yang ingin diupdate
ID_LEVEL	body	String	Yes	Berisikan id level yang ingin diubah datanya
QUESTION	body	String	Yes	Berisikan pertanyaan dari exercise tersebut
SCORE_WEIGHT	body	String	Yes	Berisikan bobot nilai untuk exercise ini.
VIDEO	body	String	No	Berisikan link video untuk exercise
AUDIO	body	file	No	Berisikan file audio untuk exercise
IMAGE	body	file	No	Berisikan file image untuk exercise
MCQ (Multiple Choice Question)				
OPTION_A	body	String	Yes	Berisikan pilihan pertama
OPTION_B	body	String	Yes	Berisikan pilihan kedua
OPTION_C	body	String	Yes	Berisikan pilihan ketiga
OPTION_D	body	String	Yes	Berisikan pilihan keempat
OPTION	body	String	Yes	Berisikan pilihan kelima

_E				
ANSWER_KEY	body	String	Yes	Jawaban dari pilihan yang telah ditambahkan
MPQ (Matching Pairs Question)				
[PAIRS][i][LEFT_PAIR]	body	Text	Yes	Pasangan kiri yang akan diinputkan ke exercise
[PAIRS][i][RIGHT_PAIR]	body	Text	Yes	Pasangan kanan yang akan diinputkan ke exercise
TFQ (True False Question)				
IS_TRUE	body	String	Yes	Berisikan jawaban benar atau salah

Tabel 126. *Parameter Update Exercise*

- Response

```
{
  "payload": {
    "ID_ADMIN_EXERCISE":
"69312f33-7d65-40c5-9600-8fc6d197a6ae",
    "ID_LEVEL": "41c20abf-fefa-4f2d-a1e9-12a881ecc241",
    "QUESTION": "Testing exercises berindex 2",
    "SCORE_WEIGHT": "6",
    "QUESTION_TYPE": "MPQ",
    "AUDIO":
"AUDIO-69312f33-7d65-40c5-9600-8fc6d197a6ae-c4775f42af4a2e1e3e4
d5cb875e261dd.mp3",
    "VIDEO":
"https://youtu.be/spMmq7j44A?si=ayVjBvoDawel0o3h",
    "IMAGE":
"IMAGE-69312f33-7d65-40c5-9600-8fc6d197a6ae-df84fe4f264501985fe
2845d0e43a19e.jpeg",
    "IS_DELETED": 0,
    "TIME_ADMIN_EXC": "2024-10-09T05:39:14.000Z",
    "matchingPairs": [
      {
        "ID_MATCHING_PAIRS":
"64a54764-3f00-4c42-b90c-aae52261aa28",
```

```

        "ID_ADMIN_EXERCISE":
"69312f33-7d65-40c5-9600-8fc6d197a6ae",
        "LEFT_PAIR": "pasangan",
        "RIGHT_PAIR": "ban 3",
        "TIME_MATCHING_PAIRS":
"2024-10-09T06:44:07.000Z"
    },
    {
        "ID_MATCHING_PAIRS":
"693b4fc0-ee8f-43e3-a488-feb263ddc584",
        "ID_ADMIN_EXERCISE":
"69312f33-7d65-40c5-9600-8fc6d197a6ae",
        "LEFT_PAIR": "ini",
        "RIGHT_PAIR": "pasangan 1",
        "TIME_MATCHING_PAIRS":
"2024-10-09T06:27:13.000Z"
    },
    {
        "ID_MATCHING_PAIRS":
"d4ab6537-8cd2-44f7-b229-a2a0c300ec75",
        "ID_ADMIN_EXERCISE":
"69312f33-7d65-40c5-9600-8fc6d197a6ae",
        "LEFT_PAIR": "pasangans 2",
        "RIGHT_PAIR": "ini",
        "TIME_MATCHING_PAIRS":
"2024-10-09T06:27:13.000Z"
    }
]
},
"statusCode": 200,
"message": "Exercise updated successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Exercise updated successfully
400	Bad Request	At least one pair is required for Matching Pairs
404	Not Found	Exercise not found

500	Server Error	Internal Server Error
-----	--------------	-----------------------

Tabel 127. *Status Code Update Exercise*

h. Update Exercise Title

Endpoint ini digunakan untuk mengubah posisi exercise untuk penggunaan 5 soal pertama harus benar nantinya. Dibawah ini adalah route serta method yang digunakan.

Route	/api/exercise/title
Method	PUT
Request	JSON raw

Tabel 128. *Route Update Exercise Title*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
ID_LEVEL	body	String	Yes	id level dari exercise yang ingin ditukarkan posisi titlenya
exercises	body	array	Yes	array dari exercise yang ingin ditukar title
Isi array exercises				
ID_ADMIN_EXERCISE	array-body	String	Yes	id dari exercise yang ingin ditukar (dengan posisi array index 0 adalah yang ingin dijadikan soal title 1 dan seterusnya untuk index lainnya)
TITLE	array-body	String	Yes	title dari exercise saat ini (dengan posisi array index 0 adalah yang ingin dijadikan soal title 1 dan seterusnya untuk

				index lainnya)
--	--	--	--	----------------

Tabel 129. *Parameter Update Exercise Title*

- Response

```
{
  "payload": {
    "updatedExercises": [
      {
        "ID_ADMIN_EXERCISE":
"391cf845-2ecf-424e-ae3b-f8948d19bb53",
        "TITLE": "Soal 1",
        "oldTitle": "Soal 2"
      },
      {
        "ID_ADMIN_EXERCISE":
"0c4f0b38-be43-4b3e-8c04-4797c3a2986b",
        "TITLE": "Soal 2",
        "oldTitle": "Soal 4"
      },
      {
        "ID_ADMIN_EXERCISE":
"c57a80eb-b3f6-42fb-af00-9a9528360a64",
        "TITLE": "Soal 4",
        "oldTitle": "Soal 1"
      },
      {
        "ID_ADMIN_EXERCISE":
"f2301bc7-f2ef-47f7-9af9-f3ef3355af78",
        "TITLE": "Soal 5",
        "oldTitle": "Soal 6"
      },
      {
        "ID_ADMIN_EXERCISE":
"7b74b8a9-a33c-407c-9adc-4347af0cea8e",
        "TITLE": "Soal 6",
        "oldTitle": "Soal 5"
      }
    ]
  },
  "statusCode": 200,
  "message": "Exercises titles swapped successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Exercises titles swapped successfully
400	Bad Request	Mismatch in exercise count
404	Not Found	Exercise with ID {ID_ADMIN_EXERCISE} not found
500	Server Error	Internal Server Error

Tabel 130. *Status Code Update Exercise Title*

- Delete Exercise File

Endpoint ini digunakan untuk menghapus file yang ada pada field unuk table exercise. Dibawah ini adalah route serta method yang digunakan.

Route	/api/exercise/file/:id
Method	DELETE
Request	JSON raw

Tabel 131. *Route Delete Exercise File*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	berisikan id dari exercise yang file atau medianya ingin dihapus
filetype	body	String	Yes	berisikan filetype dari media yang ingin dihapus (audio, image, dan video)

Tabel 132. *Parameter Delete Exercise File*

- Response

```
{
  "payload": {
    "ID_ADMIN_EXERCISE": "6275319a-1fe6-4551-8045-72a48883ae2c",
    "ID_LEVEL": "57b57bb5-101e-458a-a433-aa4954b625e6",
    "TITLE": "Soal 1",
    "QUESTION": "Apakah Naresh itu pratista?",
    "SCORE_WEIGHT": 2,
    "QUESTION_TYPE": "MCQ",
    "AUDIO": null,
    "VIDEO": null,
    "IMAGE": null,
    "TIME_ADMIN_EXC": "2024-09-05T02:51:41.000Z"
  },
  "statusCode": 200,
  "message": "Audio file deleted successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	file deleted successfully
400	Bad Request	Invalid file type specified
404	Not Found	Exercise not found
500	Server Error	Internal Server Error

Tabel 133. *Status Code Delete Exercise File*

j. Delete Exercise

Endpoint ini digunakan untuk menghapus exercise yang ada pada sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/exercise/:id
Method	DELETE

Request	JSON raw
---------	----------

Tabel 134. *Route Delete Exercise*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari exercise yang ingin dihapus

Tabel 135. *Parameter Delete Exercise*

- Response

```
{
  "payload": null,
  "statusCode": 200,
  "message": "Exercise soft-deleted successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Exercise soft-deleted successfully
404	Not Found	Exercise not found
500	Server Error	Internal Server Error

Tabel 136. *Status Code Delete Exercise*

7. StdLearning Model

Pada StdLearning Model, berisi endpoint yang berkaitan dengan manipulasi data stdLearning dalam sistem. Koleksi API ini mencakup fungsi seperti *Get StdLearnings*, *Get StdLearning by Id*, *Get Latest StdLearning Data by Level Id*, *Get Score and Next Level by StdLearning Id*, *Get Learning History*, *Get Learning*

History by Section Id, Get Learning History by Topic Id, Get Recent Student Activities, Get Recent Student Activities by Class Id, Create StdLearning, dan Student Feedback by Id StdLearning.

a. Get StdLearnings

Endpoint ini digunakan untuk mendapatkan seluruh data stdLearning yang ada. Dibawah ini adalah route serta method yang digunakan.

Route	/api/stdLearning
Method	GET
Request	JSON raw

Tabel 137. *Route Get StdLearnings*

- Parameters

Pada endpoint ini, tidak perlu mengirimkan parameter apapun, cukup memanggil route beserta methodnya yang nantinya akan divalidasi oleh sistem backend.

- Response

```
{
  "payload": [
    {
      "ID_STUDENT_LEARNING":
"ebc9edcb-732f-4003-9ef1-1e06708c8ec9",
      "ID": "26350116-9451-4f56-9581-42493db2389d",
      "ID_LEVEL": "57b57bb5-101e-458a-a433-aa4954b625e6",
      "STUDENT_START": "2024-09-09T06:43:45.000Z",
      "STUDENT_FINISH": "2024-09-09T07:10:58.000Z",
      "SCORE": 90,
      "IS_PASS": 0,
      "NEXT_LEARNING":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
      "FEEDBACK_STUDENT": null,
      "TIME_LEARNING": "2024-09-09T06:43:45.000Z"
    }
  ],
  "statusCode": 200,
  "message": "Success"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Internal Server Error

Tabel 138. *Status Code Get StdLearnings*

b. Get StdLearning by Id

Endpoint ini digunakan untuk mendapatkan data stdLearning yang ada berdasarkan id. Dibawah ini adalah route serta method yang digunakan.

Route	/api/stdLearning/:id
Method	GET
Request	JSON raw

Tabel 139. *Route Get StdLearning by Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari stdLearning yang ingin di tampilkan.

Tabel 140. *Parameter Get StdLearning by Id*

- Response

```
{
  "payload": {
    "ID_STUDENT_LEARNING": "ebc9edcb-732f-4003-9ef1-1e06708c8ec9",
    "ID": "26350116-9451-4f56-9581-42493db2389d",
    "ID_LEVEL": "57b57bb5-101e-458a-a433-aa4954b625e6",
    "STUDENT_START": "2024-09-09T06:43:45.000Z",
    "STUDENT_FINISH": "2024-09-09T07:10:58.000Z",
    "SCORE": 90,
    "IS_PASS": 0,
  }
}
```

```

        "NEXT_LEARNING": "0f89df44-edf8-40f6-a81b-1bc77628d798",
        "FEEDBACK_STUDENT": null,
        "TIME_LEARNING": "2024-09-09T06:43:45.000Z"
    },
    "statusCode": 200,
    "message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
404	Not Found	Learning not found
500	Server Error	Internal Server Error

Tabel 141. Status Get StdLearning by Id

c. Get Latest StdLearning Data by Level Id

Endpoint ini digunakan untuk mendapatkan data stdLearning paling terbaru yang ada berdasarkan id level. Dibawah ini adalah route serta method yang digunakan.

Route	/api/stdLearning/level/:levelId
Method	GET
Request	JSON raw

Tabel 142. Route Get Latest StdLearning Data by Level Id

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
levelId	params	String	Yes	berisikan level id yang stdLearningnya ingin ditampilkan

Tabel 143. *Parameter Get Latest StdLearning Data by Level Id*

- Response

```
{
  "payload": {
    "ID_STUDENT_LEARNING":
    "a61dde2f-d82f-43b5-a69f-e842000677d6",
    "ID": "26350116-9451-4f56-9581-42493db2389d",
    "ID_LEVEL": "41c20abf-feffa-4f2d-a1e9-12a881ecc241",
    "STUDENT_START": "2024-09-25T03:51:59.000Z",
    "STUDENT_FINISH": null,
    "SCORE": null,
    "IS_PASS": 0,
    "NEXT_LEARNING": null,
    "FEEDBACK_STUDENT": null,
    "TIME_LEARNING": "2024-09-25T03:51:59.000Z"
  },
  "statusCode": 200,
  "message": "Success"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
401	Unauthorized	User not authenticated
400	Bad Request	Level ID is required
404	Not Found	No StdLearning data found
500	Server Error	Internal Server Error

Tabel 144. *Status Code Get Latest StdLearning Data by Level Id*

d. Get Score and Next Level by StdLearning Id

Endpoint ini digunakan untuk mendapatkan data nilai dan level selanjutnya dari hasil pengerjaan siswa yang ada berdasarkan id stdLearning. Dibawah ini adalah route serta method yang digunakan.

Route	/api/stdLearning/score/:stdLearningId
Method	GET
Request	JSON raw

Tabel 145. *Route Get Score and Next Level by StdLearning Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
stdLearningId	params	String	Yes	berisikan id stdLearning yang ingin ditampilkan nilai dan data lainnya

Tabel 146. *Parameter Get Score and Next Level by StdLearning Id*

- Response

```
{
  "payload": {
    "ID_STUDENT_LEARNING":
"6b81668b-42e2-4254-af44-74063ac61ece",
    "NAME_USERS": "Putra",
    "NISN": 2141720074,
    "ID_LEVEL": "0f89df44-edf8-40f6-a81b-1bc77628d798",
    "CURRENT_LEVEL_NAME": "Level 6",
    "SCORE": 94,
    "NEXT_LEARNING":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
    "NEXT_LEARNING_NAME": "Level 6",
    "IS_PASS": 1
  },
  "statusCode": 200,
  "message": "Success"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
401	Unauthorized	User not authenticated
404	Not Found	No matching learning data found for this user
500	Server Error	Internal Server Error

Tabel 147. *Status Code Get Score and Next Level by StdLearning Id*

e. Get Learning History

Endpoint ini digunakan untuk mendapatkan data hasil belajar siswa untuk semua kategori section, topic, dan level. Dibawah ini adalah route serta method yang digunakan.

Route	/api/learningHistory?page=&limit=
Method	GET
Request	JSON raw

Tabel 148. *Route Get Learning History*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
page	params	int	No	page yang ingin ditampilkan.
limit	params	int	No	limit dari banyaknya data yang akan ditampilkan pada sebuah page (default adalah 10).

Tabel 149. *Parameter Get Learning History*

- Response

```
{
  "status": "success",
  "message": "Data berhasil diambil",
  "data": [
    {
      "id": 1,
      "score": 85,
      "next_level": "Lanjut ke level berikutnya"
    },
    {
      "id": 2,
      "score": 72,
      "next_level": "Ulangi level ini"
    }
  ]
}
```

```

"payload": {
  "history": [
    {
      "ID_STUDENT_LEARNING":
"7b870f3a-15aa-470c-992d-f05d8e0d722a",
      "SCORE": 91,
      "CURRENT_LEVEL": "Pretest",
      "NEXT_LEVEL": "Level 6",
      "STUDENT_FINISH": "2024-10-15T07:24:23.000Z",
      "TOPIC_NAME": "The Legend of Naresh",
      "SECTION_NAME": "Reading",
      "IS_PASS": 0
    },
    {
      "ID_STUDENT_LEARNING":
"d9184fae-065f-4720-8ba6-e8b925b30de3",
      "SCORE": 81,
      "CURRENT_LEVEL": "Pretest",
      "NEXT_LEVEL": "Level 5",
      "STUDENT_FINISH": "2024-10-14T08:39:39.000Z",
      "TOPIC_NAME": "The Legend of Naresh",
      "SECTION_NAME": "Reading",
      "IS_PASS": 0
    },
    {
      "ID_STUDENT_LEARNING":
"35fa4ea5-3d6e-4e6b-a227-a77e431a2ad1",
      "SCORE": 63,
      "CURRENT_LEVEL": "Pretest",
      "NEXT_LEVEL": "Level 3",
      "STUDENT_FINISH": "2024-10-14T08:30:05.000Z",
      "TOPIC_NAME": "The Legend of Naresh",
      "SECTION_NAME": "Reading",
      "IS_PASS": 0
    },
    {
      "ID_STUDENT_LEARNING":
"36874cfa-1010-42a7-af3d-792e544e3c66",
      "SCORE": 33,
      "CURRENT_LEVEL": "Pretest",
      "NEXT_LEVEL": "Level 1",
      "STUDENT_FINISH": "2024-10-14T07:25:53.000Z",
      "TOPIC_NAME": "The Legend of Naresh",
      "SECTION_NAME": "Reading",

```



```

        "IS_PASS": 0
      },
      {
        "ID_STUDENT_LEARNING":
"6c2888ef-e39a-4bad-a126-64e203ac6c54",
        "SCORE": 46,
        "CURRENT_LEVEL": "Pretest",
        "NEXT_LEVEL": "Level 1",
        "STUDENT_FINISH": "2024-10-14T06:15:39.000Z",
        "TOPIC_NAME": "The Legend of Naresh",
        "SECTION_NAME": "Reading",
        "IS_PASS": 0
      }
    ],
    "currentPage": 1,
    "totalPages": 2,
    "totalItems": 9
  },
  "statusCode": 200,
  "message": "Learning history retrieved successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Learning history retrieved successfully
401	Unauthorized	User not authenticated
500	Server Error	Internal Server Error

Tabel 150. Status Code Get Learning History

f. Get Learning History by Section Id

Endpoint ini digunakan untuk mendapatkan data hasil belajar siswa berdasarkan id section yang diberikan. Dibawah ini adalah route serta method yang digunakan.

Route	/api/learningHistory/section/:sectionId?page=&limit= =
-------	---

Method	GET
Request	JSON raw

Tabel 151. *Route Get Learning History by Section Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
sectionId	params	String	Yes	id dari section yang ingin ditampilkan hasil pengerjaan siswanya
page	params	int	No	page yang ingin ditampilkan.
limit	params	int	No	limit dari banyaknya data yang akan ditampilkan pada sebuah page (default adalah 10).

Tabel 152. *Parameter Get Learning History by Section Id*

- Response

```
{
  "payload": {
    "history": [
      {
        "ID_STUDENT_LEARNING":
"89fd705e-12f2-4e06-a873-0f4e09cf796f",
        "SCORE": 10,
        "CURRENT_LEVEL": "Level 4",
        "NEXT_LEVEL": "Level 4",
        "STUDENT_FINISH": "2024-10-10T08:30:31.000Z",
        "TOPIC_NAME": "Belajar Memahami Pratista",
        "SECTION_NAME": "Vocabulary",
        "IS_PASS": 0
      },
      {
        "ID_STUDENT_LEARNING":
"6b81668b-42e2-4254-af44-74063ac61ece",
```

```

        "SCORE": 94,
        "CURRENT_LEVEL": "Level 6",
        "NEXT_LEVEL": "Level 6",
        "STUDENT_FINISH": "2024-10-09T08:18:24.000Z",
        "TOPIC_NAME": "Belajar Memahami Pratista",
        "SECTION_NAME": "Vocabulary",
        "IS_PASS": 1
    },
    {
        "ID_STUDENT_LEARNING": "12341",
        "SCORE": 95,
        "CURRENT_LEVEL": "Pretest",
        "NEXT_LEVEL": "Level 6",
        "STUDENT_FINISH": "2024-10-02T02:44:53.000Z",
        "TOPIC_NAME": "Belajar Memahami Pratista",
        "SECTION_NAME": "Vocabulary",
        "IS_PASS": 0
    },
    {
        "ID_STUDENT_LEARNING":
"ebc9edcb-732f-4003-9ef1-1e06708c8ec9",
        "SCORE": 90,
        "CURRENT_LEVEL": "Level 6",
        "NEXT_LEVEL": "Level 6",
        "STUDENT_FINISH": "2024-09-09T07:10:58.000Z",
        "TOPIC_NAME": "Belajar Memahami Naresh",
        "SECTION_NAME": "Vocabulary",
        "IS_PASS": 1
    }
],
"currentPage": 1,
"totalPages": 1,
"totalItems": 4
},
"statusCode": 200,
"message": "Learning history retrieved successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Learning history retrieved

		successfully
401	Unauthorized	User not authenticated
500	Server Error	Internal Server Error

Tabel 153. *Status Code Get Learning History by Section Id*

g. Get Learning History by Topic Id

Endpoint ini digunakan untuk mendapatkan data pengerjaan siswa berdasarkan id topic. Dibawah ini adalah route serta method yang digunakan.

Route	/api/learningHistory/topic/:topicId?page=&limit=
Method	GET
Request	JSON raw

Tabel 154. *Route Get Learning History by Topic Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
topicId	params	String	Yes	id dari topic yang ingin ditampilkan hasil pengerjaan siswanya
page	params	int	No	page yang ingin ditampilkan.
limit	params	int	No	limit dari banyaknya data yang akan ditampilkan pada sebuah page (default adalah 10).

Tabel 155. *Parameter Get Learning History by Topic Id*

- Response

```
{
  "payload": {
    "history": [
```

```

{
    "ID_STUDENT_LEARNING":
"89fd705e-12f2-4e06-a873-0f4e09cf796f",
    "SCORE": 10,
    "CURRENT_LEVEL": "Level 4",
    "NEXT_LEVEL": "Level 4",
    "STUDENT_FINISH": "2024-10-10T08:30:31.000Z",
    "TOPIC_NAME": "Belajar Memahami Pratista",
    "SECTION_NAME": "Vocabulary",
    "IS_PASS": 0
},
{
    "ID_STUDENT_LEARNING":
"6b81668b-42e2-4254-af44-74063ac61ece",
    "SCORE": 94,
    "CURRENT_LEVEL": "Level 6",
    "NEXT_LEVEL": "Level 6",
    "STUDENT_FINISH": "2024-10-09T08:18:24.000Z",
    "TOPIC_NAME": "Belajar Memahami Pratista",
    "SECTION_NAME": "Vocabulary",
    "IS_PASS": 1
},
{
    "ID_STUDENT_LEARNING": "12341",
    "SCORE": 95,
    "CURRENT_LEVEL": "Pretest",
    "NEXT_LEVEL": "Level 6",
    "STUDENT_FINISH": "2024-10-02T02:44:53.000Z",
    "TOPIC_NAME": "Belajar Memahami Pratista",
    "SECTION_NAME": "Vocabulary",
    "IS_PASS": 0
}
],
"currentPage": 1,
"totalPages": 1,
"totalItems": 3
},
"statusCode": 200,
"message": "Learning history retrieved successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Learning history retrieved successfully
400	Bad Request	Topic ID is required
401	Unauthorized	User not authenticated
500	Server Error	Internal Server Error

Tabel 156. *Status Code Get Learning History by Topic Id*

h. Get Recent Student Activities

Endpoint ini digunakan untuk mendapatkan data hasil pengerjaan siswa yang diurutkan berdasarkan terakhir submit. Dibawah ini adalah route serta method yang digunakan.

Route	/api/stdLearning/activities?page=&limit=&search=
Method	GET
Request	JSON raw

Tabel 157. *Route Get Recent Student Activities*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
page	params	int	No	page yang ingin ditampilkan.
limit	params	int	No	limit dari banyaknya data yang akan ditampilkan pada sebuah page (default adalah 5).
search	params	String	No	parameter untuk melakukan search data.

Tabel 158. *Parameter Get Recent Student Activities*

- Response

```
{
  "payload": {
    "studentActivities": [
      {
        "ID_STUDENT_LEARNING":
"71f2db6b-b58c-4ad5-9b0f-45a2fd67b19a",
        "NISN": 3333311280,
        "NAME_USERS": "resh1",
        "NAME_CLASS": null,
        "NAME_SECTION": "Reading",
        "NAME_TOPIC": "The Legend of Naresh",
        "NAME_LEVEL": "Level 1",
        "SCORE": 46
      },
      {
        "ID_STUDENT_LEARNING":
"d2d92966-f93e-4af1-9a25-9535d4c32040",
        "NISN": 3333311280,
        "NAME_USERS": "resh1",
        "NAME_CLASS": null,
        "NAME_SECTION": "Reading",
        "NAME_TOPIC": "The Legend of Naresh",
        "NAME_LEVEL": "Pretest",
        "SCORE": 70
      },
      {
        "ID_STUDENT_LEARNING":
"9325803e-d402-42fa-8d40-00a798e0473b",
        "NISN": 2142154224,
        "NAME_USERS": "cobaya4",
        "NAME_CLASS": null,
        "NAME_SECTION": "Reading",
        "NAME_TOPIC": "The Legend of Naresh",
        "NAME_LEVEL": "Level 1",
        "SCORE": 100
      },
      {
        "ID_STUDENT_LEARNING":
"d5f37f4f-1bb1-416b-abbe-389a6a328588",
        "NISN": 2142154224,
        "NAME_USERS": "cobaya4",
        "NAME_CLASS": null,
        "NAME_SECTION": "Reading",
```

```

        "NAME_TOPIC": "The Legend of Naresh",
        "NAME_LEVEL": "Level 2",
        "SCORE": 43
    },
    {
        "ID_STUDENT_LEARNING":
"9fa751c9-7e45-4f8e-886f-5d1f622c7249",
        "NISN": 2142154224,
        "NAME_USERS": "cobaya4",
        "NAME_CLASS": null,
        "NAME_SECTION": "Reading",
        "NAME_TOPIC": "The Legend of Naresh",
        "NAME_LEVEL": "Pretest",
        "SCORE": 51
    }
],
    "currentPage": 1,
    "totalPages": 9,
    "totalItems": 43
},
    "statusCode": 200,
    "message": "Recent student activities retrieved
successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Recent student activities retrieved successfully
500	Server Error	Internal Server Error

Tabel 159. *Status Code Get Recent Student Activities*

i. Get Recent Student Activities by Class Id

Endpoint ini digunakan untuk mendapatkan data hasil pengerjaan siswa yang diurutkan berdasarkan terakhir submit yang di filter dengan kelas. Dibawah ini adalah route serta method yang digunakan.

Route	/api/stdLearning/activities/class/:idClass?page=1&limit=5&search=
-------	---

Method	GET
Request	JSON raw

Tabel 160. *Route Get Recent Student Activities by Class Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
idClass	params	String	Yes	id class yang ingin ditampilkan aktivitas muridnya
page	params	int	No	page yang ingin ditampilkan.
limit	params	int	No	limit dari banyaknya data yang akan ditampilkan pada sebuah page (default adalah 5).
search	params	String	No	parameter untuk melakukan search data.

Tabel 161. *Parameter Get Recent Student Activities by Class Id*

- Response

```
{
  "payload": {
    "studentActivities": [
      {
        "ID_STUDENT_LEARNING":
"7b870f3a-15aa-470c-992d-f05d8e0d722a",
        "NISN": 2141720074,
        "NAME_USERS": "Putra",
        "NAME_CLASS": "Kelas A",
        "NAME_SECTION": "Reading",
        "NAME_TOPIC": "The Legend of Naresh",
        "NAME_LEVEL": "Pretest",
        "SCORE": 91
      },
    ],
  }
}
```

```

{
    "ID_STUDENT_LEARNING":
"d9184fae-065f-4720-8ba6-e8b925b30de3",
    "NISN": 2141720074,
    "NAME_USERS": "Putra",
    "NAME_CLASS": "Kelas A",
    "NAME_SECTION": "Reading",
    "NAME_TOPIC": "The Legend of Naresh",
    "NAME_LEVEL": "Pretest",
    "SCORE": 81
},
{
    "ID_STUDENT_LEARNING":
"35fa4ea5-3d6e-4e6b-a227-a77e431a2ad1",
    "NISN": 2141720074,
    "NAME_USERS": "Putra",
    "NAME_CLASS": "Kelas A",
    "NAME_SECTION": "Reading",
    "NAME_TOPIC": "The Legend of Naresh",
    "NAME_LEVEL": "Pretest",
    "SCORE": 63
},
{
    "ID_STUDENT_LEARNING":
"36874cfa-1010-42a7-af3d-792e544e3c66",
    "NISN": 2141720074,
    "NAME_USERS": "Putra",
    "NAME_CLASS": "Kelas A",
    "NAME_SECTION": "Reading",
    "NAME_TOPIC": "The Legend of Naresh",
    "NAME_LEVEL": "Pretest",
    "SCORE": 33
},
{
    "ID_STUDENT_LEARNING":
"6c2888ef-e39a-4bad-a126-64e203ac6c54",
    "NISN": 2141720074,
    "NAME_USERS": "Putra",
    "NAME_CLASS": "Kelas A",
    "NAME_SECTION": "Reading",
    "NAME_TOPIC": "The Legend of Naresh",
    "NAME_LEVEL": "Pretest",
    "SCORE": 46
}

```

```

    ],
    "currentPage": 1,
    "totalPages": 2,
    "totalItems": 10
  },
  "statusCode": 200,
  "message": "Recent student activities retrieved successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Recent student activities retrieved successfully
404	Not Found	Class not found
500	Server Error	Internal Server Error

Tabel 162. *Status Code Get Recent Student Activities by Class Id*

j. Create StdLearning

Endpoint ini digunakan untuk membuat data stdLearning baru ke sistem.

Dibawah ini adalah route serta method yang digunakan.

Route	/api/stdLearning
Method	POST
Request	JSON raw

Tabel 163. *Route Create StdLearning*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
ID_LEVEL	body	String	Yes	id level yang ingin dibuatkan data

				stdLearning baru
--	--	--	--	------------------

Tabel 164. *Parameter Create StdLearning*

- Response

```
{
  "payload": {
    "ID_STUDENT_LEARNING":
"89fd705e-12f2-4e06-a873-0f4e09cf796f",
    "STUDENT_START": "2024-09-30T03:15:29.773Z",
    "STUDENT_FINISH": null,
    "SCORE": null,
    "IS_PASS": 0,
    "NEXT_LEARNING": null,
    "FEEDBACK_STUDENT": null,
    "TIME_LEARNING": "2024-09-30T03:15:29.773Z",
    "ID": "26350116-9451-4f56-9581-42493db2389d",
    "ID_LEVEL": "09fde9ab-bc8c-4d76-9e6c-f4f27a0f2878"
  },
  "statusCode": 201,
  "message": "Student Learning data created successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Student Learning data found and reused
201	Created	Student Learning data created successfully
401	Unauthorized	User not authenticated
404	Not Found	Level not found
500	Server Error	Internal Server Error

Tabel 165. *Status Code Create StdLearning*

k. Student Feedback by Id StdLearning

Endpoint ini digunakan untuk menambahkan feedback ke stdLearning. Dibawah ini adalah route serta method yang digunakan.

Route	/api/stdLearning/:id
Method	PUT
Request	JSON raw

Tabel 166. Route Student Feedback by Id StdLearning

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari stdLearning yang ingin ditambahkan feedback murid
FEEDBACK_STUDENT	body	String	Yes	Feedback dari murid yang akan disimpan pada stdLearning

Tabel 167. Parameter Student Feedback by Id StdLearning

- Response

```
{
  "payload": {
    "ID_STUDENT_LEARNING":
"79efcada-7567-4195-a2a7-f3e37c1305dc",
    "ID": "8c3466f1-35cb-470f-aa84-16c52fd3db8e",
    "ID_LEVEL": "0f89df44-edf8-40f6-a81b-1bc77628d798",
    "STUDENT_START": "2024-09-26T09:05:59.000Z",
    "STUDENT_FINISH": "2024-10-09T08:24:02.257Z",
    "SCORE": 94,
    "IS_PASS": 1,
    "NEXT_LEARNING":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
    "FEEDBACK_STUDENT": "mantapos mantos",
    "TIME_LEARNING": "2024-09-26T09:05:59.000Z"
  },
  "statusCode": 200,
}
```

```

"message": "Student Learning record updated successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Student Learning record updated successfully
403	Forbidden	Unauthorized to update this record
404	Not Found	Student Learning record not found
500	Server Error	Internal Server Error

Tabel 168. *Status Code Student Feedback by Id StdLearning*

8. StdExercise Model

Pada StdExercise Model, berisi endpoint yang berkaitan dengan manipulasi data stdExercise dalam sistem. Koleksi API ini mencakup fungsi seperti *Get StdExercises*, *Get StdExercise by Id*, *Get Student Answer by StdLearning Id*, dan *Student Answers*.

a. Get StdExercises

Endpoint ini digunakan untuk mendapatkan seluruh data stdExercise yang ada.

Dibawah ini adalah route serta method yang digunakan.

Route	/api/stdExercise
Method	GET
Request	JSON raw

Tabel 169. *Route Get StdExercises*

- Parameters

Pada endpoint ini, tidak perlu mengirimkan parameter apapun, cukup memanggil route beserta methodnya yang nantinya akan divalidasi oleh sistem backend.

- Response

```
{
  "payload": [
    {
      "ID_STUDENT_EXERCISE":
"5e1acb5e-3aff-42e9-93c4-bcbe900c04da",
      "ID_ADMIN_EXERCISE":
"5320711a-9c0f-4545-a2b9-7ddf3a605947",
      "ID_STUDENT_LEARNING":
"ebc9edcb-732f-4003-9ef1-1e06708c8ec9",
      "ANSWER_STUDENT": "Naresh-Pratista,
Pratista-Senggani, Senggani-Pratista",
      "IS_CORRECT": 1,
      "RESULT_SCORE_STUDENT": 2.66667,
      "TIME_STUDENT_EXC": "2024-09-09T06:49:44.000Z"
    },
    {
      "ID_STUDENT_EXERCISE":
"6ed26677-75ff-40f2-9b43-d52d82542efc",
      "ID_ADMIN_EXERCISE":
"6275319a-1fe6-4551-8045-72a48883ae2c",
      "ID_STUDENT_LEARNING":
"ebc9edcb-732f-4003-9ef1-1e06708c8ec9",
      "ANSWER_STUDENT": "D",
      "IS_CORRECT": 1,
      "RESULT_SCORE_STUDENT": 2,
      "TIME_STUDENT_EXC": "2024-09-09T06:44:47.000Z"
    },
    {
      "ID_STUDENT_EXERCISE":
"729f7448-111c-49dd-abf6-2f5c55631725",
      "ID_ADMIN_EXERCISE":
"cdde675a-237d-4f78-94f4-89ede823bbc8",
      "ID_STUDENT_LEARNING":
"ebc9edcb-732f-4003-9ef1-1e06708c8ec9",
      "ANSWER_STUDENT": "E",
      "IS_CORRECT": 1,
      "RESULT_SCORE_STUDENT": 2,
      "TIME_STUDENT_EXC": "2024-09-09T07:08:39.000Z"
    },
    {
      "ID_STUDENT_EXERCISE":
"b92c135d-a5f4-4eb3-99f5-ddb440be095a",
```

```

        "ID_ADMIN_EXERCISE":
"9a435e16-28ea-4d29-93f3-e33e9a580638",
        "ID_STUDENT_LEARNING":
"ebc9edcb-732f-4003-9ef1-1e06708c8ec9",
        "ANSWER_STUDENT": "E",
        "IS_CORRECT": 1,
        "RESULT_SCORE_STUDENT": 2,
        "TIME_STUDENT_EXC": "2024-09-09T07:10:58.000Z"
    },
    {
        "ID_STUDENT_EXERCISE":
"be89a46b-fe61-4440-b0e8-b4032c838c84",
        "ID_ADMIN_EXERCISE":
"0ac4b6b3-21d6-4000-88d7-8d9b432f9668",
        "ID_STUDENT_LEARNING":
"ebc9edcb-732f-4003-9ef1-1e06708c8ec9",
        "ANSWER_STUDENT": "1",
        "IS_CORRECT": 1,
        "RESULT_SCORE_STUDENT": 2,
        "TIME_STUDENT_EXC": "2024-09-09T06:47:33.000Z"
    },
    {
        "ID_STUDENT_EXERCISE":
"e10e6a0d-4bd7-4918-aa28-446c2cab4d26",
        "ID_ADMIN_EXERCISE":
"42cbd79f-ad94-404c-9c9d-51b77b414ade",
        "ID_STUDENT_LEARNING":
"ebc9edcb-732f-4003-9ef1-1e06708c8ec9",
        "ANSWER_STUDENT": "0",
        "IS_CORRECT": 1,
        "RESULT_SCORE_STUDENT": 2,
        "TIME_STUDENT_EXC": "2024-09-09T07:10:17.000Z"
    }
],
"statusCode": 200,
"message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success

500	Server Error	Internal Server Error
-----	--------------	-----------------------

Tabel 170. *Status Code Get StdExercises*

b. Get StdExercise by Id

Endpoint ini digunakan untuk mendapatkan data stdExercise yang ada berdasarkan id. Dibawah ini adalah route serta method yang digunakan.

Route	/api/stdExercise/:id
Method	GET
Request	JSON raw

Tabel 171. *Route Get StdExercise by Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari stdExercise yang ingin di tampilkan.

Tabel 172. *Parameter Get StdExercise by Id*

- Response

```
{
  "payload": {
    "ID_STUDENT_EXERCISE": "5e1acb5e-3aff-42e9-93c4-bcbe900c04da",
    "ID_ADMIN_EXERCISE": "5320711a-9c0f-4545-a2b9-7ddf3a605947",
    "ID_STUDENT_LEARNING": "ebc9edcb-732f-4003-9ef1-1e06708c8ec9",
    "ANSWER_STUDENT": "Naresh-Pratista, Pratista-Senggani, Senggani-Pratista",
    "IS_CORRECT": 1,
    "RESULT_SCORE_STUDENT": 2.66667,
    "TIME_STUDENT_EXC": "2024-09-09T06:49:44.000Z"
  },
  "statusCode": 200,
  "message": "Success"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
404	Not Found	stdExercise not found
500	Server Error	Internal Server Error

Tabel 173. Status Code Get StdExercise by Id

c. Get Student Answer by StdLearning Id

Endpoint ini digunakan untuk mendapatkan data jawaban siswa yang ada berdasarkan id dari stdLearning. Dibawah ini adalah route serta method yang digunakan.

Route	/api/studentAnswers/:id
Method	GET
Request	JSON raw

Tabel 174. Route Get Student Answer by StdLearning Id

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari stdLearning yang jawaban siswanya ingin ditampilkan.

Tabel 175. Parameter Get Student Answer by StdLearning Id

- Response

```
{
  "payload": {
    "ID_STUDENT_LEARNING":
    "79efcada-7567-4195-a2a7-f3e37c1305dc",
```

```

        "ID_LEVEL": "0f89df44-edf8-40f6-a81b-1bc77628d798",
        "NAME_SECTION": "Vocabulary",
        "NAME_TOPIC": "Belajar Memahami Pratista",
        "NAME_LEVEL": "Level 6",
        "SCORE": 94,
        "IS_PASS": 1,
        "stdExercises": [
            {
                "ID_STUDENT_EXERCISE": null,
                "ID_ADMIN_EXERCISE":
"15b4d936-24a7-4842-af1f-6e8ac9f35ab9",
                "TITLE": "Soal 1",
                "QUESTION": "Apakah Naresh itu pratista?",
                "QUESTION_TYPE": "MCQ",
                "SCORE_WEIGHT": 2,
                "IMAGE": null,
                "VIDEO": null,
                "AUDIO":
"audio-15b4d936-24a7-4842-af1f-6e8ac9f35ab9-1f3433d8d7ba85e911f
62a93bb002b5a.mp3",
                "ANSWER_STUDENT": null,
                "IS_CORRECT": null,
                "RESULT_SCORE_STUDENT": null,
                "multipleChoices": [
                    {
                        "OPTION_A": "bukan",
                        "OPTION_B": "sangat bukan",
                        "OPTION_C": "no way",
                        "OPTION_D": "naresh adalah senggani",
                        "OPTION_E": "senggani adalah naresh"
                    }
                ]
            },
            {
                "ID_STUDENT_EXERCISE":
"c56923fc-aec3-4718-90e9-abcc87ee0863",
                "ID_ADMIN_EXERCISE":
"1643d97d-1095-45ab-91a8-38748431fb0b",
                "TITLE": "Soal 2",
                "QUESTION": "Apakah Naresh itu pratista?",
                "QUESTION_TYPE": "TFQ",
                "SCORE_WEIGHT": 2,
                "IMAGE": null,

```

```

"VIDEO":
"https://youtu.be/07Ll4UH_PgM?si=MHNu67zsc4Z0j3_f",
    "AUDIO": null,
    "ANSWER_STUDENT": "0",
    "IS_CORRECT": 1,
    "RESULT_SCORE_STUDENT": 2
  },
  {
    "ID_STUDENT_EXERCISE":
"585e072e-8c68-4f7c-8943-29f2e7344463",
    "ID_ADMIN_EXERCISE":
"5a52b52f-ad04-4652-b9c3-6f9c213111c9",
    "TITLE": "Soal 3",
    "QUESTION": "Siapa saja yang termasuk pada
semanggi man?",
    "QUESTION_TYPE": "MPQ",
    "SCORE_WEIGHT": 2,
    "IMAGE": null,
    "VIDEO":
"https://youtu.be/07Ll4UH_PgM?si=MHNu67zsc4Z0j3_f",
    "AUDIO": null,
    "ANSWER_STUDENT": "Naresh-Senggani,
Pratista-Senggani, Senggani-Pratista",
    "IS_CORRECT": 1,
    "RESULT_SCORE_STUDENT": 1.33333,
    "matchingPairs": [
      {
        "LEFT_PAIR": "Senggani",
        "RIGHT_PAIR": "Naresh"
      },
      {
        "LEFT_PAIR": "Senggani",
        "RIGHT_PAIR": "Pratista"
      }
    ]
  },
  {
    "ID_STUDENT_EXERCISE":
"c35a7dad-5761-4368-942c-5a4fe85fb7e7",
    "ID_ADMIN_EXERCISE":
"4019e25d-e187-4946-b50a-32258b9f5c3b",
    "TITLE": "Soal 4",
    "QUESTION": "Apakah Pratista itu Naresh?",
    "QUESTION_TYPE": "MCQ",

```

```

        "SCORE_WEIGHT": 2,
        "IMAGE": null,
        "VIDEO":
"https://youtu.be/07Ll4UH_PgM?si=MHNu67zsc4Z0j3_f",
        "AUDIO": null,
        "ANSWER_STUDENT": "E",
        "IS_CORRECT": 1,
        "RESULT_SCORE_STUDENT": 2,
        "multipleChoices": [
            {
                "OPTION_A": "bukan",
                "OPTION_B": "sangat bukan",
                "OPTION_C": "no way",
                "OPTION_D": "naresh adalah senggani",
                "OPTION_E": "senggani adalah naresh"
            }
        ]
    },
    {
        "ID_STUDENT_EXERCISE":
"b6dabb8c-4b03-4f36-972e-94971534d19d",
        "ID_ADMIN_EXERCISE":
"7502cdfd-4237-4307-941a-63e0403b6c31",
        "TITLE": "Soal 5",
        "QUESTION": "Apakah Naresh itu Senggani?",
        "QUESTION_TYPE": "TFQ",
        "SCORE_WEIGHT": 2,
        "IMAGE": null,
        "VIDEO": null,
        "AUDIO":
"audio-7502cdfd-4237-4307-941a-63e0403b6c31-1f3433d8d7ba85e911f62a93bb002b5a.mp3",
        "ANSWER_STUDENT": "1",
        "IS_CORRECT": 1,
        "RESULT_SCORE_STUDENT": 2
    },
    {
        "ID_STUDENT_EXERCISE":
"98a0e3d1-fc4d-4760-bbb1-3190515b521a",
        "ID_ADMIN_EXERCISE":
"58f59acf-5940-4912-bc7c-2cf7b18ac17b",
        "TITLE": "Soal 6",
        "QUESTION": "Apakah Pratista itu Senggani?",
        "QUESTION_TYPE": "TFQ",

```

```

        "SCORE_WEIGHT": 2,
        "IMAGE": null,
        "VIDEO": null,
        "AUDIO": null,
        "ANSWER_STUDENT": "1",
        "IS_CORRECT": 1,
        "RESULT_SCORE_STUDENT": 2
    }
]
},
"statusCode": 200,
"message": "Student learning exercises retrieved successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Student learning exercises retrieved successfully
404	Not Found	Student learning data not found
500	Server Error	Error retrieving student learning exercises

Tabel 176. *Status Code Get Student Answer by StdLearning Id*

d. Student Answers

Endpoint ini digunakan untuk menginputkan jawaban siswa ke sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/stdExercise
Method	POST
Request	JSON raw

Tabel 177. *Route Get Student Answers*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
answers	body	array	Yes	array untuk menginputkan jawaban siswa ke sistem
Isi array asnwrs				
ID_STUDENT_LEARNING	body	String	Yes	berisi id student learning saat ini yang akan diinputkan dengan jawaban siswa
ID_ADMIN_EXERCISE	body	String	Yes	id exercise yang dikerjakan oleh siswa
ANSWER_STUDENT	body	String	Yes	jawaban untuk soal tersebut

Tabel 178. *Parameter Get Student Answers*

- Response

```
{
  "payload": {
    "ID_STUDENT_LEARNING":
"79efcada-7567-4195-a2a7-f3e37c1305dc",
    "ID": "8c3466f1-35cb-470f-aa84-16c52fd3db8e",
    "ID_LEVEL": "0f89df44-edf8-40f6-a81b-1bc77628d798",
    "STUDENT_START": "2024-09-26T09:05:59.000Z",
    "STUDENT_FINISH": "2024-09-26T09:14:06.676Z",
    "SCORE": 94,
    "IS_PASS": 1,
    "NEXT_LEARNING":
"0f89df44-edf8-40f6-a81b-1bc77628d798",
    "FEEDBACK_STUDENT": null,
    "TIME_LEARNING": "2024-09-26T09:05:59.000Z",
    "MONITORING": {
      "ID_MONITORING":
"7b12352f-0090-424c-9737-f12881f60532",
      "TIME_MONITORING": "2024-09-26T09:14:06.693Z",
    }
  }
}
```

```

"ID_STUDENT_LEARNING":
"79efcada-7567-4195-a2a7-f3e37c1305dc",
  "ID_CLASS": null
},
"statusCode": 200,
"message": "Student Learning record updated and monitoring
created successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
400	Bad Request	{data} is required
404	Not Found	Id student learning not found
500	Server Error	Error creating or updating student exercise data!

Tabel 179. Status Code Get Student Answers

9. Class Model

Pada Class Model, berisi endpoint yang berkaitan dengan manipulasi data Class dalam sistem. Koleksi API ini mencakup fungsi seperti *Get Classes*, *Get Class by Id*, *Get Class for Admin*, *Get Student By Class Id*, *Get Student With No Class*, *Create Class*, *Update Student Class By Name and NISN*, *Update Class*, dan *Delete Class*.

a. Get Classes

Endpoint ini digunakan untuk mendapatkan seluruh data class yang ada. Dibawah ini adalah route serta method yang digunakan.

Route	/api/classes
Method	GET
Request	JSON raw

Tabel 180. Route Get Classes

- Parameters

Pada endpoint ini, tidak perlu mengirimkan parameter apapun, cukup memanggil route beserta methodnya yang nantinya akan divalidasi oleh sistem backend.

- Response

```
{
  "payload": [
    {
      "ID_CLASS": "a5911b15-c105-4d78-ad39-c4051ca97510",
      "NAME_CLASS": "Kelas B",
      "TOTAL_STUDENT": 32,
      "TIME_CLASS": null
    },
    {
      "ID_CLASS": "e1387a79-58d8-4ac1-8e8d-656fc5cd3ef2",
      "NAME_CLASS": "Kelas A",
      "TOTAL_STUDENT": 33,
      "TIME_CLASS": null
    }
  ],
  "statusCode": 200,
  "message": "Success"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Internal Server Error

Tabel 181. *Status Code Get Classes*

b. Get Class by Id

Endpoint ini digunakan untuk mendapatkan data Class yang ada berdasarkan id. Dibawah ini adalah route serta method yang digunakan.

Route	/api/class/:id
Method	GET

Request	JSON raw
---------	----------

Tabel 182. *Route Get Class by Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari class yang ingin di tampilkan.

Tabel 183. *Parameter Get Class by Id*

- Response

```
{
  "payload": {
    "ID_CLASS": "a5911b15-c105-4d78-ad39-c4051ca97510",
    "NAME_CLASS": "Kelas B",
    "TOTAL_STUDENT": 32,
    "TIME_CLASS": null
  },
  "statusCode": 200,
  "message": "Success"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
404	Not Found	Class not found
500	Server Error	Internal Server Error

Tabel 184. *Status Code Get Class by Id*

c. Get Class for Admin

Endpoint ini digunakan untuk mendapatkan data Class yang ada pada sistem untuk halaman admin. Dibawah ini adalah route serta method yang digunakan.

Route	/api/class/admin?page=&limit=&search=&sort=
Method	GET
Request	JSON raw

Tabel 185. *Route Get Class for Admin*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
page	params	int	No	berisikan page yang ingin ditampilkan dengan nilai default adalah 1
limit	params	int	No	berisikan banyak data yang ditampilkan pada sebuah page dengan nilai default adalah 1
search	params	String	No	digunakan untuk melakukan search data pada payload
sort	params	String	No	dilakukan untuk melakukan sorting

Tabel 186. *Parameter Get Class for Admin*

- Response

```
{
  "payload": {
    "classes": [
      {
        "ID_CLASS": "e1387a79-58d8-4ac1-8e8d-656fc5cd3ef2",
        "NAME_CLASS": "Kelas A",
        "STUDENTS": 3,
        "TOTAL_STUDENT": 33,
      }
    ]
  }
}
```

```

        "TIME_CLASS": null
    },
    {
        "ID_CLASS":
"a5911b15-c105-4d78-ad39-c4051ca97510",
        "NAME_CLASS": "Kelas B",
        "STUDENTS": 0,
        "TOTAL_STUDENT": 32,
        "TIME_CLASS": null
    },
    {
        "ID_CLASS":
"1c2898b7-7577-4ada-9674-a7afa04b5a85",
        "NAME_CLASS": "Kelas F",
        "STUDENTS": 0,
        "TOTAL_STUDENT": 33,
        "TIME_CLASS": "2024-10-10T06:17:28.000Z"
    }
],
"currentPage": 1,
"totalPages": 1,
"totalItems": 3
},
"statusCode": 200,
"message": "Classes retrieved successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Classes retrieved successfully
500	Server Error	Error retrieving classes data!

Tabel 187. Status Code Get Class for Admin

d. Get Student by Class Id

Endpoint ini digunakan untuk mendapatkan data student yang berkaitan dengan suatu class. Dibawah ini adalah route serta method yang digunakan.

Route	/api/class/student/:classId
-------	-----------------------------

Method	GET
Request	JSON raw

Tabel 188. *Route Get Student by Class Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
classId	params	String	Yes	Berisikan id class yang ingin ditampilkan siswanya

Tabel 189. *Parameter Get Student by Class Id*

- Response

```
{
  "payload": [
    {
      "ID": "fee4e191-7eea-4503-a64a-c0f1b255c2d5",
      "NAME_USERS": "elangs",
      "EMAIL": "elangs@gmail.com",
      "NISN": 2141720073,
      "NAME_CLASS": "Kelas A",
      "ROLE": "student"
    },
    {
      "ID": "7351d439-f42a-4184-b6d5-e05dbab08c85",
      "NAME_USERS": "elangptra",
      "EMAIL": "elangptra17@gmail.com",
      "NISN": 2141720077,
      "NAME_CLASS": "Kelas A",
      "ROLE": "student"
    },
    {
      "ID": "26350116-9451-4f56-9581-42493db2389d",
      "NAME_USERS": "Putra",
      "EMAIL": "putra@gmail.com",
      "NISN": 2141720074,
      "NAME_CLASS": "Kelas A",
    }
  ]
}
```

```

        "ROLE": "student"
      }
    ],
    "statusCode": 200,
    "message": "Success"
  }
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Error retrieving students by class ID

Tabel 190. *Status Code Get Student by Class Id*

e. Get Student With No Class

Endpoint ini digunakan untuk mendapatkan data student yang belum memiliki class terkait. Dibawah ini adalah route serta method yang digunakan.

Route	/api/user/student/unassigned
Method	GET
Request	JSON raw

Tabel 191. *Route Get Student With No Class*

- Parameters

Pada endpoint ini, tidak perlu mengirimkan parameter apapun, cukup kirimkan endpoint dan backend akan mengembalikan payload data jika terdapat data student.

- Response

```

{
  "payload": [
    {
      "ID": "96f6a0c7-dcb1-4014-be14-85e94b107957",
      "NAME_USERS": "elangsa",
      "EMAIL": "elangsa@gmail.com",

```

```

        "NISN": 2141720076,
        "NAME_CLASS": null,
        "ROLE": "student"
    },
    {
        "ID": "b9303295-1694-40d2-a440-72b8cbe788f2",
        "NAME_USERS": "Elang aceh ajaa",
        "EMAIL": "elang12@gmail.com",
        "NISN": 2181518194,
        "NAME_CLASS": null,
        "ROLE": "student"
    },
    {
        "ID": "a50c09c2-289f-43b0-a6ef-6a98a8de1314",
        "NAME_USERS": "Maguires",
        "EMAIL": "maguire@gmail.com",
        "NISN": 6419184865,
        "NAME_CLASS": null,
        "ROLE": "student"
    },
    {
        "ID": "347183f9-4e17-4d07-86c6-54e5037c4236",
        "NAME_USERS": "ranti",
        "EMAIL": "ranti20@gmail.com",
        "NISN": 784643210,
        "NAME_CLASS": null,
        "ROLE": "student"
    },
    {
        "ID": "8c3466f1-35cb-470f-aa84-16c52fd3db8e",
        "NAME_USERS": "elangs",
        "EMAIL": "elangsz@gmail.com",
        "NISN": 214172012,
        "NAME_CLASS": null,
        "ROLE": "student"
    },
    {
        "ID": "5b8234cc-000f-4c77-ad3d-ab211cceeff1",
        "NAME_USERS": "elang",
        "EMAIL": "elangptra16@gmail.com",
        "NISN": 1345484845,
        "NAME_CLASS": null,
        "ROLE": "student"
    },

```

```

{
  "ID": "ec621b2d-9587-4518-a094-97ab9492a813",
  "NAME_USERS": "Agoyy",
  "EMAIL": "agoy@gmail.com",
  "NISN": 2141700574,
  "NAME_CLASS": null,
  "ROLE": "student"
},
{
  "ID": "581dabd4-9272-4088-9fe9-650ca269e6d5",
  "NAME_USERS": "Samir",
  "EMAIL": "samir@gmail.com",
  "NISN": 3124846464,
  "NAME_CLASS": null,
  "ROLE": "student"
},
{
  "ID": "d523371d-9836-4a62-9cb3-4ea34b914e78",
  "NAME_USERS": "testa",
  "EMAIL": "test@gmail.com",
  "NISN": 4343461816,
  "NAME_CLASS": null,
  "ROLE": "student"
}
],
"statusCode": 200,
"message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Error retrieving students with no class

Tabel 192. *Status Code Get Student With No Class*

f. Create Class

Endpoint ini digunakan untuk membuat data class baru pada sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/class
Method	POST
Request	JSON raw

Tabel 193. *Route Create Class*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
NAME_CLASS	body	String	Yes	nama dari kelas yang ingin dibuat
TOTAL_STUDENT	body	int	Yes	jumlah murid yang diharapkan ada pada satu kelas (jumlah murid bisa lebih dari total_student)

Tabel 194. *Parameter Create Class*

- Response

```
{
  "payload": {
    "ID_CLASS": "3b93e40c-10b0-425b-90fa-7dde2948184b",
    "TIME_CLASS": "2024-12-11T03:35:40.571Z",
    "NAME_CLASS": "Kelas C",
    "TOTAL_STUDENT": "35"
  },
  "statusCode": 201,
  "message": "Class created successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
201	Created	Class created successfully
400	Bad Request	{data} is required

500	Server Error	Internal Server Error
-----	--------------	-----------------------

Tabel 195. *Status Code Create Class*

g. Update Student Class by Name and NISN

Endpoint ini digunakan untuk melakukan update data student dan mengaitkan student tersebut dengan sebuah class. Dibawah ini adalah route serta method yang digunakan.

Route	/api/class/student/update
Method	POST
Request	JSON raw

Tabel 196. *Route Update Student Class by Name and NISN*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
NAME_CLASS	body	String	Yes	berisikan nama kelas yang ingin dikaitkan dengan data student
STUDENTS	body	array	Yes	merupakan array dari student yang ingin dikaitkan dengan class
Isi array STUDENTS				
NAME_USERS	array-body	String	Yes	nama dari user yang ingin dikaitkan dengan class
NISN	array-body	bigint	Yes	nisn dari user tersebut

Tabel 197. *Parameter Update Student Class by Name and NISN*

- Response

```
{
  "message": "Success"
}
```

```

"payload": {
  "updateResults": [
    {
      "NAME_USERS": "elangs",
      "NISN": "2141720073",
      "message": "Student's class and related
monitoring updated successfully",
      "studentUpdateResults": []
    },
    {
      "NAME_USERS": "elangptra",
      "NISN": "2141720077",
      "message": "Student's class and related
monitoring updated successfully",
      "studentUpdateResults": [
        {
          "ID_MONITORING":
"63624b73-5ba9-4f6e-a8b8-48ed8c167040",
          "ID_STUDENT_LEARNING":
"93811aa8-0ea8-46d9-a264-c146f8f0c503",
          "ID_CLASS":
"e1387a79-58d8-4ac1-8e8d-656fc5cd3ef2",
          "ID_GURU": null,
          "FEEDBACK_GURU": null,
          "TIME_MONITORING": null
        }
      ]
    }
  ],
  "statusCode": 200,
  "message": "All students updated successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	All students updated successfully
400	Bad Request	{data} is required
404	Not Found	Class not found

500	Server Error	Internal Server Error
-----	--------------	-----------------------

Tabel 198. *Status Code Update Student Class by Name and NISN*

h. Update Class

Endpoint ini digunakan untuk melakukan update data Class yang ada berdasarkan id dari class tersebut. Dibawah ini adalah route serta method yang digunakan.

Route	/api/class/update/:id
Method	PUT
Request	JSON raw

Tabel 199. *Route Update Class*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari class yang ingin diupdate
NAME_CLASS	body	String	No	nama baru untuk class yang ingin diubah saat ini
TOTAL_STUDENT	body	int	No	jumlah murid yang diharapkan pada satu kelas tersebut (jumlah siswa bisa lebih dari total student)

Tabel 200. *Parameter Update Class*

- Response

```
{
  "payload": {
    "ID_CLASS": "3b93e40c-10b0-425b-90fa-7dde2948184b",
    "NAME_CLASS": "Kelas D",
```

```

    "TOTAL_STUDENT": 33,
    "TIME_CLASS": "2024-12-11T03:35:40.000Z"
  },
  "statusCode": 200,
  "message": "Class updated successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Class updated successfully
400	Bad Request	Class with this name already exists
404	Not Found	Class not found
500	Server Error	Internal Server Error

Tabel 201. *Status Code Update Class*

- Delete Class

Endpoint ini digunakan untuk menghapus data Class yang ada pada sistem berdasarkan id. Dibawah ini adalah route serta method yang digunakan.

Route	/api/class/delete/:id
Method	DELETE
Request	JSON raw

Tabel 202. *Route Delete Class*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari class yang ingin dihapus dari sistem

Tabel 203. *Parameter Delete Class*

- Response

```
{
  "payload": null,
  "statusCode": 200,
  "message": "Class deleted successfully and related models updated"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Class deleted successfully and related models updated
404	Not Found	Class not found
500	Server Error	Internal Server Error

Tabel 204. *Status Code Delete Class*

10. Monitoring Model

Pada Monitoring Model, berisi endpoint yang berkaitan dengan manipulasi data monitoring dalam sistem. Koleksi API ini mencakup fungsi seperti *Get Monitorings*, *Get Monitoring by Id*, *Monitoring Student Progress*, *Monitoring Student Progress by Id*, *Download Monitoring Student Progress to CSV by Monitoring Id*, *Monitoring Feedback*, *Get Monitoring Data by Topic Id*, *Get Class Monitoring by Class Id*, *Get Class Monitoring by Class & Topic Id*, *Monitoring Feedback by Class*, dan *Download Class Monitoring Data to CSV by Class and Topic Id*.

a. Get Monitorings

Endpoint ini digunakan untuk mendapatkan seluruh data monitoring yang ada. Dibawah ini adalah route serta method yang digunakan.

Route	/api/monitoring
Method	GET

Request	JSON raw
---------	----------

Tabel 205. *Route Get Monitorings*

- Parameters

Pada endpoint ini, tidak perlu mengirimkan parameter apapun, cukup memanggil route beserta methodnya yang nantinya akan divalidasi oleh sistem backend.

- Response

```
{
  "payload": [
    {
      "ID_MONITORING":
"141fa9e5-f4bc-4da2-aa46-3096be1f05bc",
      "ID_STUDENT_LEARNING":
"79efcada-7567-4195-a2a7-f3e37c1305dc",
      "ID_CLASS": null,
      "ID_GURU": null,
      "FEEDBACK_GURU": null,
      "TIME_MONITORING": "2024-09-26T09:20:38.000Z"
    },
    {
      "ID_MONITORING":
"63624b73-5ba9-4f6e-a8b8-48ed8c167040",
      "ID_STUDENT_LEARNING":
"93811aa8-0ea8-46d9-a264-c146f8f0c503",
      "ID_CLASS": "e1387a79-58d8-4ac1-8e8d-656fc5cd3ef2",
      "ID_GURU": null,
      "FEEDBACK_GURU": null,
      "TIME_MONITORING": null
    },
    {
      "ID_MONITORING":
"a06b7689-b4ae-4913-a4e0-b246c48d9735",
      "ID_STUDENT_LEARNING":
"6b81668b-42e2-4254-af44-74063ac61ece",
      "ID_CLASS": "e1387a79-58d8-4ac1-8e8d-656fc5cd3ef2",
      "ID_GURU": "ba06663f-22bd-40b1-b5bc-d6c189d362fe",
      "FEEDBACK_GURU": "Selamat! Kamu sudah menyelesaikan
quest menjadi Monkey King!",
      "TIME_MONITORING": null
    }
  ]
}
```

```

],
"statusCode": 200,
"message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Internal Server Error

Tabel 206. *Status Code Get Monitorings*

b. Get Monitoring by Id

Endpoint ini digunakan untuk mendapatkan data monitoring yang ada berdasarkan id. Dibawah ini adalah route serta method yang digunakan.

Route	/api/monitoring/:id
Method	GET
Request	JSON raw

Tabel 207. *Route Get Monitoring by Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari monitoring yang ingin di tampilkan.

Tabel 208. *Parameter Get Monitoring by Id*

- Response

```

{
  "payload": {
    "ID_MONITORING": "a06b7689-b4ae-4913-a4e0-b246c48d9735",
    "ID_STUDENT_LEARNING": "6b81668b-42e2-4254-af44-74063ac61ece",

```



```

        "ID_CLASS": "e1387a79-58d8-4ac1-8e8d-656fc5cd3ef2",
        "ID_GURU": "ba06663f-22bd-40b1-b5bc-d6c189d362fe",
        "FEEDBACK_GURU": "Selamat! Kamu sudah menyelesaikan quest
menjadi Monkey King!",
        "TIME_MONITORING": null
    },
    "statusCode": 200,
    "message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
404	Not Found	Monitoring not found
500	Server Error	Internal Server Error

Tabel 209. *Status Code Get Monitoring by Id*

c. Monitoring Student Progress

Endpoint ini digunakan untuk mendapatkan data pembelajaran siswa yang ada pada sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/monitoring/progress?page=&limit=&search=&sort=
Method	GET
Request	JSON raw

Tabel 210. *Route Monitoring Student Progress*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
page	params	int	No	berisikan halaman yang ingin ditampilkan

limit	params	int	No	jumlah dari data yang ingin ditampilkan pada tiap halaman
search	params	String	No	untuk melakukan search data yang ada pada payload
sort	params	String	No	untuk melakukan sorting data

Tabel 211. *Parameter Monitoring Student Progress*

- Response

```
{
  "payload": {
    "monitorings": [
      {
        "ID_MONITORING": "14129402",
        "NISN": 2141720074,
        "NAME_USERS": "Putra",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Naresh",
        "NAME_CLASS": "Kelas A"
      },
      {
        "ID_MONITORING":
"a06b7689-b4ae-4913-a4e0-b246c48d9735",
        "NISN": 2141720074,
        "NAME_USERS": "Putra",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Pratista",
        "NAME_CLASS": "Kelas A"
      },
      {
        "ID_MONITORING":
"63624b73-5ba9-4f6e-a8b8-48ed8c167040",
        "NISN": 2141720077,
        "NAME_USERS": "elangptra",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Pratista",
        "NAME_CLASS": "Kelas A"
      },
      {

```

```

"ID_MONITORING":
"45752ce6-46b8-4802-8045-e7990af54609",
  "NISN": 214172012,
  "NAME_USERS": "elangs",
  "NAME_SECTION": "Listening",
  "NAME_TOPIC": "Belajar Memahami Pratista",
  "NAME_CLASS": null
},
"currentPage": 1,
"totalPages": 1,
"totalItems": 4
},
"statusCode": 200,
"message": "Success retrieving student monitoring
progress!"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success retrieving student monitoring progress!
500	Server Error	Error retrieving student monitoring progress!

Tabel 212. *Status Code Monitoring Student Progress*

d. Monitoring Student Progress by Id

Endpoint ini digunakan untuk mendapatkan data hasil pembelajaran siswa secara lengkap yang ada pada sistem berdasarkan id. Dibawah ini adalah route serta method yang digunakan.

Route	/api/monitoring/progress/:id?page=&limit=&search=&sort=
Method	GET
Request	JSON raw

Tabel 213. *Route Monitoring Student Progress by Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari monitoring yang ada pada sistem
page	params	int	No	berisikan halaman yang ingin ditampilkan
limit	params	int	No	jumlah dari data yang ingin ditampilkan pada tiap halaman
search	params	String	No	untuk melakukan search data yang ada pada payload
sort	params	String	No	untuk melakukan sorting data

Tabel 214. *Parameter Monitoring Student Progress by Id*

- Response

```
{
  "payload": {
    "ID_MONITORING":
"a06b7689-b4ae-4913-a4e0-b246c48d9735",
    "NAME_SECTION": "Listening",
    "NAME_TOPIC": "Belajar Memahami Pratista",
    "NAME_USERS": "Putra",
    "NISN": 2141720074,
    "levels": [
      {
        "NAME_LEVEL": "Level 6",
        "SCORE": 94,
        "FEEDBACK_STUDENT": "mantap",
        "STUDENT_START": "2024-09-12T04:15:54.000Z",
        "STUDENT_FINISH": "2024-10-09T08:18:24.000Z"
      },
      {
        "NAME_LEVEL": "Level 4",
        "SCORE": 53,
```

```

        "FEEDBACK_STUDENT": null,
        "STUDENT_START": "2024-09-30T04:00:30.000Z",
        "STUDENT_FINISH": "2024-10-10T09:19:52.000Z"
    },
    {
        "NAME_LEVEL": "Level 4",
        "SCORE": 10,
        "FEEDBACK_STUDENT": null,
        "STUDENT_START": "2024-09-30T03:15:29.000Z",
        "STUDENT_FINISH": "2024-10-10T09:18:23.000Z"
    }
],
"currentPage": 1,
"totalPages": 1,
"totalItems": 3
},
"statusCode": 200,
"message": "Success retrieving student progress!"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success retrieving student progress!
500	Server Error	Error retrieving student progress!

Tabel 215. *Status Code Monitoring Student Progress by Id*

e. Download Monitoring Student Progress to CSV by Monitoring Id

Endpoint ini digunakan untuk melakukan download hasil belajar siswa yang ada pada sistem berdasarkan id monitoring. Dibawah ini adalah route serta method yang digunakan.

Route	/api/monitoring/progress/csv/:id
Method	GET
Request	JSON raw

Tabel 216. *Route Download Monitoring Student Progress to CSV by Monitoring Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	berisikan id monitoring yang data pembelajaran siswanya ingin diunduh

Tabel 217. *Parameter Download Monitoring Student Progress to CSV by Monitoring Id*

- Response

Response akan berisikan sebuah file excel dalam format .csv yang akan dikirimkan melalui backend ke client.

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
404	Not Found	Monitoring not found
500	Server Error	Error retrieving student progress!

Tabel 218. *Status Code Download Monitoring Student Progress to CSV by Monitoring Id*

f. Monitoring Feedback

Endpoint ini digunakan untuk memberikan feedback pada hasil pengerjaan siswa yang telah menyelesaikan sebuah topic yang ada pada sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/monitoring/feedback/:id
Method	POST
Request	JSON raw

Tabel 219. *Route Monitoring Feedback*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	berisikan id monitoring belajar siswa
feedback	body	String	Yes	berisikan feedback yang akan diberikan pada siswa

Tabel 220. *Parameter Monitoring Feedback*

- Response

```
{
  "payload": {
    "ID_MONITORING": "a06b7689-b4ae-4913-a4e0-b246c48d9735",
    "FEEDBACK_GURU": "Selamat! Kamu sudah menyelesaikan quest menjadi Monkey King!",
    "ID_GURU": "ba06663f-22bd-40b1-b5bc-d6c189d362fe",
    "TEACHER_NAME": "Prateachta"
  },
  "statusCode": 200,
  "message": "Success updating teacher feedback!"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success updating teacher feedback!
401	Unauthorized	User not authenticated
404	Not Found	Teacher not found!
500	Server Error	Error updating teacher feedback!

Tabel 221. *Status Code Monitoring Feedback*

g. Get Monitoring Data by Topic Id

Endpoint ini digunakan untuk mendapatkan data monitoring beserta feedback guru yang telah diberikan dengan berdasarkan topic id (dengan catatan user telah login pada sistem). Dibawah ini adalah route serta method yang digunakan.

Route	/api/monitoring/topic/:topicId
Method	GET
Request	JSON raw

Tabel 222. *Route Get Monitoring Data by Topic Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	berisikan id dari topic yang ingin dilihat data monitoringnya

Tabel 223. *Parameter Get Monitoring Data by Topic Id*

- Response

```
{
  "payload": {
    "ID_MONITORING":
    "a06b7689-b4ae-4913-a4e0-b246c48d9735",
    "NAME_TOPIC": "Belajar Memahami Pratista",
    "NAME_LEVEL": "Level 6",
    "TEACHER_NAME": "Prateachta",
    "FEEDBACK_GURU": "Selamat! Kamu sudah menyelesaikan
quest menjadi Monkey King!",
    "TIME_MONITORING": null
  },
  "statusCode": 200,
  "message": "Success retrieving monitoring data!"
}
```



```
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success retrieving monitoring data!
401	Unauthorized	User not authenticated
500	Server Error	Error retrieving monitoring data!

Tabel 224. *Status Code Get Monitoring Data by Topic Id*

h. Get Class Monitoring by Class Id

Endpoint ini digunakan untuk mendapatkan data monitoring yang ada pada sistem dengan berdasarkan class id yang sama. Dibawah ini adalah route serta method yang digunakan.

Route	/api/monitoring/class/:classId?page=&limit=&search=&sort=
Method	GET
Request	JSON raw

Tabel 225. *Route Get Class Monitoring by Class Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
classId	params	String	Yes	class id yang ingin ditampilkan seluruh data monitoringnya
page	params	int	No	berisikan halaman yang ingin ditampilkan
limit	params	int	No	jumlah dari data yang

				ingin ditampilkan pada tiap halaman
search	params	String	No	untuk melakukan search data yang ada pada payload
sort	params	String	No	untuk melakukan sorting data

Tabel 226. *Parameter Get Class Monitoring by Class Id*

- Response

```
{
  "payload": {
    "data": [
      {
        "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
        "ID_TOPIC":
"c2453b27-c9df-4f24-8502-ae32bd8044ad",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Naresh"
      },
      {
        "ID_SECTION":
"49390272-efbd-4f1c-9b63-39fa8905b891",
        "ID_TOPIC":
"891f6b2e-b385-4dc5-9bbd-97a44daf7138",
        "NAME_SECTION": "Listening",
        "NAME_TOPIC": "Belajar Memahami Pratista"
      }
    ],
    "currentPage": 1,
    "totalPages": 1,
    "totalItems": 2
  },
  "statusCode": 200,
  "message": "Success retrieving unique monitoring section
and topic data!"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success retrieving unique monitoring section and topic data!
500	Server Error	Error retrieving monitoring section and topic data!

Tabel 227. *Status Code Get Class Monitoring by Class Id*

i. Get Class Monitoring by Class & Topic Id

Endpoint ini digunakan untuk mendapatkan data monitoring yang ada pada sistem dengan berdasarkan class id dan topic id. Dibawah ini adalah route serta method yang digunakan.

Route	/api/monitoring/class?page=&limit=&=&sort=
Method	POST
Request	JSON raw

Tabel 228. *Route Get Class Monitoring by Class & Topic Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
page	params	int	No	berisikan halaman yang ingin ditampilkan
limit	params	int	No	jumlah dari data yang ingin ditampilkan pada tiap halaman
search	params	String	No	untuk melakukan search data yang ada pada payload
sort	params	String	No	untuk melakukan sorting data
ID_CLA	body	String	Yes	id class yang ingin

SS				ditampilkan data monitoringnya
ID_TOPI C	body	String	Yes	id topic yang ingin ditampilkan data monitoringnya

Tabel 229. *Parameter Get Class Monitoring by Class & Topic Id*

- Response

```
{
  "payload": {
    "ID_CLASS": "e1387a79-58d8-4ac1-8e8d-656fc5cd3ef2",
    "ID_SECTION": "49390272-efbd-4f1c-9b63-39fa8905b891",
    "ID_TOPIC": "891f6b2e-b385-4dc5-9bbd-97a44daf7138",
    "NAME_CLASS": "Kelas A",
    "NAME_SECTION": "Vocabulary",
    "NAME_TOPIC": "Belajar Memahami Pratista",
    "levels": [
      {
        "NISN": 2141720077,
        "NAME_USERS": "elangptra",
        "NAME_LEVEL": "Level 6",
        "SCORE": 89,
        "FEEDBACK_STUDENT": null,
        "STUDENT_START": "2024-09-11T07:02:38.000Z",
        "STUDENT_FINISH": "2024-09-12T04:17:48.000Z"
      },
      {
        "NISN": 2141720074,
        "NAME_USERS": "Putra",
        "NAME_LEVEL": "Level 6",
        "SCORE": 94,
        "FEEDBACK_STUDENT": "mantap",
        "STUDENT_START": "2024-09-12T04:15:54.000Z",
        "STUDENT_FINISH": "2024-10-09T08:18:24.000Z"
      },
      {
        "NISN": 2141720074,
        "NAME_USERS": "Putra",
        "NAME_LEVEL": "Level 4",
        "SCORE": 10,
        "FEEDBACK_STUDENT": null,
        "STUDENT_START": "2024-09-30T03:15:29.000Z",

```

```

        "STUDENT_FINISH": "2024-10-10T08:30:31.000Z"
      },
      {
        "NISN": 2141720074,
        "NAME_USERS": "Putra",
        "NAME_LEVEL": "Pretest",
        "SCORE": 95,
        "FEEDBACK_STUDENT": null,
        "STUDENT_START": "2024-10-02T01:44:53.000Z",
        "STUDENT_FINISH": "2024-10-02T02:44:53.000Z"
      }
    ],
    "currentPage": 1,
    "totalPages": 1,
    "totalItems": 4
  },
  "statusCode": 200,
  "message": "Success retrieving monitoring data!"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success retrieving monitoring data!
404	Not Found	{data} not found
500	Server Error	Error retrieving monitoring data!

Tabel 230. *Status Code Get Class Monitoring by Class & Topic Id*

j. Monitoring Feedback by Class

Endpoint ini digunakan untuk memberikan feedback terhadap seluruh murid yang telah menyelesaikan topic dengan berdasarkan class dan topic. Dibawah ini adalah route serta method yang digunakan.

Route	/api/monitoring/feedback/class
Method	POST
Request	JSON raw

Tabel 231. *Route Get Monitoring Feedback by Class*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
ID_CLASS	body	String	Yes	id class yang ingin diberikan feedback
ID_TOPIC	body	String	Yes	topic yang berkaitan
FEEDBACK	body	String	Yes	feedback dari guru

Tabel 232. *Parameter Get Monitoring Feedback by Class*

- Response

```
{
  "payload": [
    {
      "ID_MONITORING":
"63624b73-5ba9-4f6e-a8b8-48ed8c167040",
      "FEEDBACK_GURU": "Selamat! Kamu sudah menyelesaikan
quest menjadi Monkey King!",
      "ID_GURU": "ba06663f-22bd-40b1-b5bc-d6c189d362fe",
      "TEACHER_NAME": "Prateachta"
    },
    {
      "ID_MONITORING":
"a06b7689-b4ae-4913-a4e0-b246c48d9735",
      "FEEDBACK_GURU": "Selamat! Kamu sudah menyelesaikan
quest menjadi Monkey King!",
      "ID_GURU": "ba06663f-22bd-40b1-b5bc-d6c189d362fe",
      "TEACHER_NAME": "Prateachta"
    }
  ],
  "statusCode": 200,
  "message": "Success updating teacher feedback for class and
topic!"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success updating teacher feedback for class and topic!
401	Unauthorized	User not authenticated
404	Not Found	Teacher not found!
500	Server Error	Error updating teacher feedback!

Tabel 233. *Status Code Get Monitoring Feedback by Class*

k. Download Class Monitoring Data to CSV by Class and Topic Id

Endpoint ini digunakan untuk mendownload data hasil pembelajaran siswa untuk satu kelas pada sebuah topic yang telah diselesaikan. Dibawah ini adalah route serta method yang digunakan.

Route	/api/monitoring/class/csv
Method	POST
Request	JSON raw

Tabel 234. *Route Download Class Monitoring Data to CSV by Class and Topic Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
ID_CLASS	body	String	Yes	id class yang data pembelajarannya ingin diunduh
ID_TOPIC	body	String	Yes	id topic yang berkaitan untuk mengunduh hasil pembelajaran

Tabel 235. *Parameter Download Class Monitoring Data to CSV by Class and Topic Id*

- Response

Response akan berisikan sebuah file excel dalam format .csv yang akan dikirimkan melalui backend ke client.

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
404	Not Found	{data} not found!
500	Server Error	Error retrieving monitoring data!

Tabel 236. *Status Code Download Class Monitoring Data to CSV by Class and Topic Id*

11. Report Model

Pada Report Model, berisi endpoint yang berkaitan dengan manipulasi data report dalam sistem. Koleksi API ini mencakup fungsi seperti *Get Reports*, *Get Five New Reports*, *Get Report By Id*, *Get Report By User Id*, dan *Create Report*.

a. Get Reports

Endpoint ini digunakan untuk mendapatkan seluruh data report yang ada. Dibawah ini adalah route serta method yang digunakan.

Route	/api/report?page=&limit=&search=&sort=
Method	GET
Request	JSON raw

Tabel 237. *Route Get Reports*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
-----------	------	-----------	----------	-------------

page	params	int	No	page yang ingin ditampilkan.
limit	params	int	No	limit dari banyaknya data yang akan ditampilkan pada sebuah page (default adalah 10)
search	params	String	No	parameter untuk melakukan search data report
sort	params	String	No	parameter untuk melakukan sort berdasarkan email, nama, ataupun report.

Tabel 238. *Parameter Get Reports*

- Response

```
{
  "payload": {
    "reports": [
      {
        "ID_REPORT": "10635aaa-177b-4df5-a71e-381ae5cb1940",
        "ID": "12027fb9-365f-485e-8b23-52e2bc3e4352",
        "REPORTS": "Naresh bukan Monkey King, aku kecewa!",
        "TIME_REPORT": "2024-09-25T03:19:46.000Z",
        "USER_NAME": "Adam",
        "USER_EMAIL": "adam@gmail.com"
      },
      {
        "ID_REPORT": "52dd879c-d69e-48d6-b4d1-8f082106d73f",
        "ID": "12027fb9-365f-485e-8b23-52e2bc3e4352",
        "REPORTS": "Kenapa Naersh malah menjadi Cu Pat Kai, bukan Monkey King. Aku sangat kecewa!",
        "TIME_REPORT": "2024-09-25T03:33:53.000Z",
        "USER_NAME": "Adam",
        "USER_EMAIL": "adam@gmail.com"
      },
      {

```

```

"ID_REPORT":
"55105721-e14b-48bb-a361-3b630cf84cf4",
      "ID": "ec621b2d-9587-4518-a094-97ab9492a813",
      "REPORTS": "butut",
      "TIME_REPORT": "2024-09-25T04:38:18.000Z",
      "USER_NAME": "Agoyy",
      "USER_EMAIL": "agoy@gmail.com"
    }
  ],
  "currentPage": 1,
  "totalPages": 1,
  "totalItems": 3
},
"statusCode": 200,
"message": "Reports retrieved successfully"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Internal Server Error

Tabel 239. *Status Code Get Reports*

b. Get Five New Reports

Endpoint ini digunakan untuk mendapatkan 5 data report terbaru yang ada pada sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/report/new
Method	GET
Request	JSON raw

Tabel 240. *Route Get Five New Reports*

- Parameters

Pada endpoint ini, tidak perlu mengirimkan parameter apapun, cukup memanggil route beserta methodnya yang nantinya akan divalidasi oleh sistem backend.

- Response

```
{
  "payload": [
    {
      "ID_REPORT":
"55105721-e14b-48bb-a361-3b630cf84cf4",
      "ID": "ec621b2d-9587-4518-a094-97ab9492a813",
      "REPORTS": "butut",
      "TIME_REPORT": "2024-09-25T04:38:18.000Z",
      "USER_NAME": "Agoyy",
      "USER_EMAIL": "agoy@gmail.com"
    },
    {
      "ID_REPORT":
"e25f87c2-2dab-4b92-8bc0-71d3527c0285",
      "ID": "ec621b2d-9587-4518-a094-97ab9492a813",
      "REPORTS": "test",
      "TIME_REPORT": "2024-09-25T04:37:51.000Z",
      "USER_NAME": "Agoyy",
      "USER_EMAIL": "agoy@gmail.com"
    },
    {
      "ID_REPORT":
"52dd879c-d69e-48d6-b4d1-8f082106d73f",
      "ID": "12027fb9-365f-485e-8b23-52e2bc3e4352",
      "REPORTS": "Kenapa Naersh malah menjadi Cu Pat Kai,
bukan Monkey King. Aku sangat kecewa!",
      "TIME_REPORT": "2024-09-25T03:33:53.000Z",
      "USER_NAME": "Adam",
      "USER_EMAIL": "adam@gmail.com"
    },
    {
      "ID_REPORT":
"10635aaa-177b-4df5-a71e-381ae5cb1940",
      "ID": "12027fb9-365f-485e-8b23-52e2bc3e4352",
      "REPORTS": "Naresh bukan Monkey King, aku kecewa!",
      "TIME_REPORT": "2024-09-25T03:19:46.000Z",
      "USER_NAME": "Adam",
      "USER_EMAIL": "adam@gmail.com"
    }
  ],
  "statusCode": 200,
  "message": "Five latest reports retrieved successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Internal Server Error

Tabel 241. *Status Code Get Five New Reports*

c. Get Report By Id

Endpoint ini digunakan untuk mendapatkan data report yang ada berdasarkan id. Dibawah ini adalah route serta method yang digunakan.

Route	/api/report/:id
Method	GET
Request	JSON raw

Tabel 242. *Route Get Report By Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari report yang ingin di tampilkan.

Tabel 243. *Parameter Get Report By Id*

- Response

```
{
  "payload": {
    "ID_REPORT": "52dd879c-d69e-48d6-b4d1-8f082106d73f",
    "ID": "12027fb9-365f-485e-8b23-52e2bc3e4352",
    "REPORTS": "Kenapa Naersh malah menjadi Cu Pat Kai, bukan Monkey King. Aku sangat kecewa!",
    "TIME_REPORT": "2024-09-25T03:33:53.000Z",
    "USER_NAME": "Adam",
  }
}
```

```

    "USER_EMAIL": "adam@gmail.com"
  },
  "statusCode": 200,
  "message": "Success"
}

```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
404	Not Found	Report not found
500	Server Error	Internal Server Error

Tabel 244. *Status Code Get Report By Id*

d. Get Report By User Id

Endpoint ini digunakan untuk mendapatkan data report yang ada berdasarkan id dari user yang terdaftar pada sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/report/user/:id
Method	GET
Request	JSON raw

Tabel 245. *Route Get Report By User Id*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
id	params	String	Yes	id dari user yang reportnya ingin ditampilkan.

Tabel 246. *Parameter Get Report By User Id*

- Response

```
{
  "payload": [
    {
      "ID_REPORT": "10635aaa-177b-4df5-a71e-381ae5cb1940",
      "ID": "12027fb9-365f-485e-8b23-52e2bc3e4352",
      "REPORTS": "Naresh bukan Monkey King, aku kecewa!",
      "TIME_REPORT": "2024-09-25T03:19:46.000Z",
      "USER_NAME": "Adam",
      "USER_EMAIL": "adam@gmail.com"
    },
    {
      "ID_REPORT": "52dd879c-d69e-48d6-b4d1-8f082106d73f",
      "ID": "12027fb9-365f-485e-8b23-52e2bc3e4352",
      "REPORTS": "Kenapa Naersh malah menjadi Cu Pat Kai, bukan Monkey King. Aku sangat kecewa!",
      "TIME_REPORT": "2024-09-25T03:33:53.000Z",
      "USER_NAME": "Adam",
      "USER_EMAIL": "adam@gmail.com"
    }
  ],
  "statusCode": 200,
  "message": "Success"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
200	Ok	Success
500	Server Error	Internal Server Error

Tabel 247. *Status Code Get Report By User Id*

e. Create Report

Endpoint ini digunakan untuk membuat data entri report ke sistem. Dibawah ini adalah route serta method yang digunakan.

Route	/api/report
-------	-------------

Method	POST
Request	JSON raw

Tabel 248. *Route Create Report*

- Parameters

Pada endpoint ini, terdapat beberapa parameter yang perlu dikirimkan ke server. Dibawah ini adalah parameter yang perlu diberikan.

Parameter	Type	Data Type	Required	Description
REPORTS	body	String	Yes	Berisikan report yang ingin diinputkan ke sistem

Tabel 249. *Parameter Create Report*

- Response

```
{
  "payload": {
    "ID_REPORT": "52dd879c-d69e-48d6-b4d1-8f082106d73f",
    "ID": "12027fb9-365f-485e-8b23-52e2bc3e4352",
    "REPORTS": "Kenapa Naersh malah menjadi Cu Pat Kai, bukan Monkey King. Aku sangat kecewa!",
    "TIME_REPORT": "2024-09-25T03:33:53.000Z",
    "USER_NAME": "Adam",
    "USER_EMAIL": "adam@gmail.com"
  },
  "statusCode": 201,
  "message": "Report created successfully"
}
```

- Errors

Endpoint ini menggunakan beberapa error code seperti dibawah ini.

Code	Error	Message
201	Created	Report created successfully
400	Bad Request	Report content is required
401	Unauthorized	User not authenticated

500	Server Error	Internal Server Error
-----	--------------	-----------------------

Tabel 250. *Status Code Create Report*