Documentação da API: Mentor-api

Esta documentação descreve os endpoints da coleção Postman "Mentor-api", agrupados por suas funcionalidades (`Users`, `Auth`, `exercises` e `lesson_plans`).

Variáveis de Ambiente Necessárias:

- **`{{api-url}}`: URL base da API (ex: `https://api.mentor.com.br`)**
- **`{{token}}`: Token de acesso JWT (Bearer Token) para endpoints autenticados.**

1. Users

Endpoints para gerenciamento de usuários.

1.1. add user

Cria um novo usuário no sistema.

Método	URL	Headers	Body (JSON Structure)
POST	{{api-url}}/users/create	Content-Type: application/json	name_user: String, email_user: String, cpf_user: String, password_user: String, role_user: String (ex: "admin", "student", "teacher")

Body Exemplo:

```
{
   "name_user": "Thiago Ferreira",
   "email_user": "thiago@thc.com",
   "cpf_user": "99091731079",
   "password_user": "123455678",
   "role_user": "admin"
}
```

Response (Status 201 - Created):

```
{
   "message": "Usuário criado com sucesso.",
   "sucess": true,
   "createdUser": {
      "id_user": "uuid",
```

```
      "name_user": "string",
      "email_user": "string",
      "cpf_user": "string",
      "role_user": "string"
   }
}
```

1.2. get users

Retorna a lista de todos os usuários.

Método	URL	Headers	Body (JSON Structure)
GET	{{api-url}}/users/get-all	Authorization: Bearer {{token}}	N/A

Response (Status 200 - OK):

```
{
   "users": [
      {
         "id_user": "uuid",
         "name_user": "string",
         "email_user": "string",
         "cpf_user": "string",
         "role_user": "string"
      }
      // ... more users
   ],
   "sucess": true
}
```

1.3. patch user

Atualiza dados do usuário (usuário pode atualizar seus próprios dados).

Método	URL	Headers	Body (JSON Structure)

PATCH	{{api-url}}/users/patch/:id_user	Authorization: Bearer {{token}}	Atributos do usuário a serem alterados (ex: name_user)

Body Exemplo:

```
{
   "name_user": "Thiago Ferreira Gonçalves"
}
```

Response (Status 200 - OK):

```
{
   "message": "Usuário atualizado com sucesso.",
   "sucess": true,
   "updatedUser": {
      "id_user": "uuid",
      "name_user": "string (updated)",
      "email_user": "string",
      "cpf_user": "string",
      "role_user": "string"
   }
}
```

1.4. patch user admin

Atualiza dados de qualquer usuário (acesso de administrador).

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/users/patchAdmin/:id_user	Authorization: Bearer {{token}}	Atributos do usuário a serem alterados (ex: name_user)

Body Exemplo:

```
{
   "name_user": "Thiago"
```

```
}
```

Response (Status 200 - OK):

Retorno idêntico ao patch user com os dados atualizados.

2. Auth

Endpoints para autenticação e gerenciamento de tokens.

2.1. login

Realiza o login e retorna os tokens de acesso.

Método	URL	Headers	Body (JSON Structure)
POST	{{api-url}}/auth/login	Content-Type: application/json	email_user: String, password_user: String

Body Exemplo:

```
{
  "email_user": "thiago@thc.com",
  "password_user": "123455678"
}
```

Response (Status 200 - OK):

```
{
  "tokens": {
    "accessToken": "jwt_token_string",
    "refreshToken": "jwt_token_string"
  },
  "user": {
    "id_user": "uuid",
    "name_user": "string",
    "email_user": "string",
    "cpf_user": "string",
    "role_user": "string"
  }
}
```

2.2. refresh token

Gera novos tokens de acesso e refresh usando o `refreshToken` expirado.

Método	URL	Headers	Body (JSON Structure)
`POST`	`{{api-url}}/auth/refresh-token`	`Content-Type: application/json`	`refreshToken`: String (refresh token JWT)

Body Exemplo:

```
{
   "refreshToken": "jwt_token_string_here"
}
```

Response (Status 200 - OK):

```
{
   "tokens": {
      "accessToken": "new_jwt_token_string",
      "refreshToken": "new_jwt_token_string"
   },
   "user": {
      "id_user": "uuid",
      "role_user": "string"
   }
}
```

3. exercises

Endpoints para gerenciamento de exercícios criados por IA ou manualmente.

3.1. add exercise

Cria um novo conjunto de exercícios.

Método	URL	Headers	Body (JSON Structure)

POST	{{api-url}}/exercises/create-exercises	Authorization: Bearer {{token}}, Content-Type: application/json	Objeto complexo contendo detalhes do exercício e sua estrutura.

Estrutura do Body (Apenas atributos principais):

```
{
  "subject_exercises": "string",
  "description_exercises": "string",
  "grade_level_exercises": "string",
  "complexity_level_exercises": "string",
  "duration_minutes_exercises": "number",
  "objectives_exercises": [
    {"titleObjectiveExercises": "string", "contentObjectiveExercises": "string"}
  ],
  "themes_exercises": [
    {"titleThemeExercises": "string", "contentThemeExercises": "string"}
  ],
  "exercises": [
    {
      "type_exercise": "multipla-escolha" / "discursiva" / "verdadeiro-falso",
      "title_exercise": "string",
      "content_exercise": "string",
      "options_exercise_multipla_escolha": [
        {"option": "string", "content_option": "string"}
      ],
      "options_exercise_verdadeiro_falso": [
        {"option": "V"/"F", "content_option": "string"}
      ],
      "correct_answer_exercise": "string (e.g., 'E', 'V, F, V, F, V')",
      "explanation_exercise": "string",
      "bloom_level": "string"
    }
  ]
}
```

Response (Status 201 - Created):

Retorna a estrutura completa do exercício criado, incluindo todos os campos e sub-objetos enviados no Body, junto com IDs gerados (id_exercise, id_exercise_item, etc.) e timestamps.

```
{
```

```
  "subject_exercises": "string",
  "description_exercises": "string",
  "grade_level_exercises": "string",
  "complexity_level_exercises": "string",
  "duration_minutes_exercises": 90,
  "objectives_exercises": [
    { /* ... */ },
    { /* ... */ }
  ],
  "themes_exercises": [
    { /* ... */ },
    { /* ... */ }
  ],
  "exercises": [
    {
      "type_exercise": "string",
      "title_exercise": "string",
      "content_exercise": "string",
      "options_exercise_multipla_escolha": [
        { /* ... */ }
      ],
      // ... demais campos com IDs e timestamps
    }
  ]
}
```

3.2. get by admin

Retorna uma lista paginada de todos os exercícios (acesso de administrador).

Método	URL	Headers	Body (JSON Structure)
GET	{{api-url}}/exercises/get-exercises/admin?page={number}&limit={number}	Authorization: Bearer {{token}}	N/A

Query Params:

- page: Número da página.
- limit: Limite de itens por página.

Response (Status 200 - OK):

```
{
```

```
  "total": "number",
  "totalPages": "number",
  "currentPage": "number",
  "data": [
    {
      "id_exercise": "uuid",
      "id_user": "uuid",
      "subjectExercises": "string",
      "descriptionExercises": "string",
      "gradeLevelExercises": "string",
      "complexityLevelExercises": "string",
      "durationMinutesExercises": "number",
      "createdAt": "date-time",
      "updatedAt": "date-time",
      "execiseItems": [
        {
          "id_exercise_item": "uuid",
          "id_exercise": "uuid",
          "type_exercise": "string",
          "title_exercise": "string",
          // ... (demais campos, incluindo optionsMultiple/optionsTrueOrFalse)
        }
      ],
      "themeExercises": [ /* ... array de objetos de temas ... */ ],
      "objectiveExercises": [ /* ... array de objetos de objetivos ... */ ]
    }
    // ... more exercises
  ]
}
```

3.3. get by user

Retorna uma lista paginada de exercícios criados pelo usuário autenticado.

Método	URL	Headers	Body (JSON Structure)
GET	{{api-url}}/exercises/get-exercises/user?page={number}&limit={number}	Authorization: Bearer {{token}}	N/A

Query Params:

- page: **Número da página.**
- limit: **Limite de itens por página.**

Response (Status 200 - OK):

Retorno idêntico ao get by admin, mas filtrado pelo id_user do token.

3.4. get by id

Retorna um exercício específico pelo seu ID.

Método	URL	Headers	Body (JSON Structure)
GET	{{api-url}}/exercises/get-exercises/exercise/{id_exercise}	Authorization: Bearer {{token}}	N/A

Path Params:

- id_exercise: ID do exercício.

Response (Status 200 - OK):

Retorna o objeto de exercício completo (estrutura do item dentro do array data do get by admin).

3.5. update ExerciseModel

Atualiza campos principais do exercício.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/exercises/update/exercise/{id_exercise}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos a serem atualizados (subjectExercises, descriptionExercises, durationMinutesExercises, etc.)

Body Exemplo:

```
{
  "subjectExercises": "Estruturas de Dados em Java (atualizado)",
```

```
  "descriptionExercises": "Revisão dos exercícios sobre listas duplamente encadeadas.",
  "durationMinutesExercises": 120
}
```

Response (Status 200 - OK):

```
{
    "updatedExercise": {
        "id_exercise": "uuid",
        "id_user": "uuid",
        "subjectExercises": "string (updated)",
        // ... demais campos do exercício atualizados
        "createdAt": "date-time",
        "updatedAt": "date-time"
    },
    "message": "Exercício atualizado com sucesso.",
    "sucess": true
}
```

3.6. update ExerciseItem

Atualiza um item/pergunta específica dentro de um exercício.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/exercises/update/exercise-item/{id_exercise_item}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos do item/pergunta a serem atualizados.

Body Exemplo:

```
{
  "title_exercise": "Afirmação sobre remoção (atualizada)",
  "explanation_exercise": "Nova explicação mais detalhada sobre remoção de nós."
}
```

Response (Status 200 - OK):

Retorna o objeto do item do exercício atualizado, aninhado com o objeto pai do exercício.

```
{
   "updatedExercise": {
      "id_exercise_item": "uuid",
      "title_exercise": "string (updated)",
      "explanation_exercise": "string (updated)",
      // ... demais campos do item do exercício
      "exercise": {
         "id_exercise": "uuid",
         // ... campos principais do exercício pai
      }
   },
   "message": "Exercício atualizado com sucesso.",
   "sucess": true
}
```

3.7. update ObjectiveExercise

Atualiza um objetivo específico do exercício.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/exercises/update/objective/{id_objective_exercises}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos do objetivo a serem atualizados (titleObjectiveExercises, contentObjectiveExercises).

Body Exemplo:

```
{
  "titleObjectiveExercises": "Compreender melhor listas duplamente encadeadas (atualizado)"
}
```

Response (Status 200 - OK):

Retorna o objeto do objetivo atualizado, aninhado com o objeto pai do exercício.

```
{
   "updatedExercise": {
      "id_objective_exercises": "uuid",
      "titleObjectiveExercises": "string (updated)",
      // ... demais campos do objetivo
      "exercise": {
         "id_exercise": "uuid",
         // ... campos principais do exercício pai
      }
   },
   "message": "Exercício atualizado com sucesso.",
   "sucess": true
}
```

3.8. update theme

Atualiza um tema específico do exercício.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/exercises/update/theme/{id_theme_exercise}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos do tema a serem atualizados (titleThemeExercises, contentThemeExercises).

Body Exemplo:

```
{
  "contentThemeExercises": "Revisão da estrutura de nós e ponteiros nas listas (atualizado)"
}
```

Response (Status 200 - OK):

Retorno idêntico ao update ObjectiveExercise, mas com o objeto id_theme_exercise atualizado.

3.9. update option-multiple

Atualiza uma opção de múltipla escolha.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/exercises/update/option-multiple/{id_optionsMultiple}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos da opção a serem atualizados (content_option, option).

Body Exemplo:

```
{
  "content_option": "Cada nó mantém dois ponteiros: um para o próximo e outro para o anterior (atualizado)"
}
```

Response (Status 200 - OK):

Retorna o objeto da opção múltipla atualizado, aninhado com o item do exercício e o exercício pai.

```
{
    "updatedExercise": {
        "id_optionsMultiple": "uuid",
        "content_option": "string (updated)",
        "exerciseItem": {
            "id_exercise_item": "uuid",
            // ... campos do item do exercício
        }
    },
    "message": "Exercício atualizado com sucesso.",
    "sucess": true
}
```

3.10. update option-true-or-false

Atualiza uma opção de verdadeiro ou falso.

Método	URL	Headers	Body (JSON Structure)

PATCH	{{api-url}}/exercises/update/option-true-or-false/{id_optionsTrueOrFalse}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos da opção a serem atualizados (content_option, option).

Body Exemplo:

```
{
  "content_option": "A afirmação é verdadeira em casos específicos (atualizado)"
}
```

Response (Status 200 - OK):

Retorno idêntico ao update option-multiple, mas com o objeto id_optionsTrueOrFalse atualizado.

4. lesson_plans

Endpoints para gerenciamento de planos de aula.

4.1. add lesson_plan

Cria um novo plano de aula.

Método	URL	Headers	Body (JSON Structure)
POST	{{api-url}}/lesson-plans/create-lesson-plan	Authorization: Bearer {{token}}, Content-Type: application/json	Objeto completo do plano de aula.

Estrutura do Body (Apenas atributos principais):

```
{
   "subjectLessonPlan": "string",
   "descriptionLessonPlan": "string",
   "gradeLevelLessonPlan": "string",
   "complexityLevelLessonPlan": "string",
   "durationMinutesLessonPlan": "number",
   "generalObjective": "string",
   "specificObjectives": [ /* array de objetos */ ],
   "competencies": [ /* array de objetos */ ],
   "themes": [ /* array de objetos */ ],
```

```
  "teachingMethodologies": [ /* array de objetos */ ],
  "topics": [ /* array complexo de objetos de tópicos */ ],
  "homework": { /* objeto de casa */ },
  "inclusiveAdaptation": { /* objeto de adaptação */ },
  "references": [ /* array de objetos */ ],
  "closure": { /* objeto de encerramento */ }
}
```

Response (Status 201 - Created):

Retorna a estrutura completa do plano de aula criado, incluindo todos os campos e sub-objetos com IDs e timestamps gerados.

```
{
  "message": "Plano de aula criado com sucesso.",
  "sucess": true,
  "createdLessonPlan": {
    "id_lesson_plan": "uuid",
    "id_user": "uuid",
    "subjectLessonPlan": "string",
    // ... demais campos
    "objetives_lesson_plan": [ /* ... com IDs gerados ... */ ],
    "competencies_lesson_plan": [ /* ... com IDs gerados ... */ ],
    // ... demais sub-objetos aninhados com seus respectivos IDs
  }
}
```

4.2. get by admin (lesson_plans)

Retorna uma lista paginada de todos os planos de aula (acesso de administrador).

Método	URL	Headers	Body (JSON Structure)
GET	{{api-url}}/lesson-plans/get-lesson-plans/admin?page={number}&limit={number}	Authorization: Bearer {{token}}	N/A

Query Params:

- page: Número da página.
- limit: Limite de itens por página.

Response (Status 200 - OK):

```
{
   "total": "number",
   "totalPages": "number",
   "currentPage": "number",
   "data": [
      {
         "id_lesson_plan": "uuid",
         "id_user": "uuid",
         "subjectLessonPlan": "string",
         // ... campos principais
         "objetives_lesson_plan": [ /* array de objetos com IDs */ ],
         "competencies_lesson_plan": [ /* array de objetos com IDs */ ],
         "themes_lesson_plan": [ /* array de objetos com IDs */ ],
         "methodology_lesson_plan": [ /* array de objetos com IDs */ ],
         "topics_lesson_plan": [ /* array de objetos complexos com sub-arrays */ ],
         // ... homework_lesson_plan, inclusive_adaptation_lesson_plan,
references_lesson_plan, closure_lesson_plan
      }
      // ... more lesson plans
   ]
}
```

4.3. get by user (lesson_plans)

Retorna uma lista paginada de planos de aula criados pelo usuário autenticado.

Método	URL	Headers	Body (JSON Structure)
GET	{{api-url}}/lesson-plans/get-lesson-plans/user?page={number}&limit={number}	Authorization: Bearer {{token}}	N/A

Query Params:

- page: Número da página.
- limit: Limite de itens por página.

Response (Status 200 - OK):

Retorno idêntico ao get by admin, mas filtrado pelo id_user do token.

4.4. get by id (lesson_plans)

Retorna um plano de aula específico pelo seu ID.

Método	URL	Headers	Body (JSON Structure)
GET	{{api-url}}/lesson-plans/get-lesson-plans/id/{id_lesson_plan}	Authorization: Bearer {{token}}	N/A

Path Params:

- id_lesson_plan: ID do plano de aula.

Response (Status 200 - OK):

Retorna o objeto de plano de aula completo (estrutura do item dentro do array data do get by admin).

4.5. patch lesson

Atualiza campos principais do plano de aula.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/lesson-plans/update/lesson-plan/{id_lesson_plan}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos principais a serem atualizados.

Body Exemplo:

```
{
   "subjectLessonPlan": "Estrutura de Dados em Java novo"
}
```

Response (Status 200 - OK):

```
{
   "message": "Plano de aula atualizado com sucesso.",
   "sucess": true,
   "updatedLessonPlan": {
      "id_lesson_plan": "uuid",
      "subjectLessonPlan": "string (updated)",
      // ... demais campos principais atualizados
   }
```

```
}
```

4.6. patch objetives

Atualiza um objetivo específico do plano de aula.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/lesson-plans/update/objetives/{id_objetives_lesson_plan}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos do objetivo a serem atualizados.

Body Exemplo:

```
{
  "titleObjetivesLessonPlan": "Compreender o conceito de lista duplamente encadeada"
}
```

Response (Status 200 - OK):

Retorna o objeto do objetivo atualizado, aninhado com o objeto pai do plano de aula.

```
{
  "message": "Plano de aula atualizado com sucesso.",
  "sucess": true,
  "updatedLessonPlan": {
    "id_objetives_lesson_plan": "uuid",
    "titleObjetivesLessonPlan": "string (updated)",
    // ... demais campos do objetivo
    "lesson_plan": {
      "id_lesson_plan": "uuid",
      // ... campos principais do plano de aula pai
    }
  }
}
```

4.7. patch competencies

Atualiza uma competência específica do plano de aula.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/lesson-plans/update/competencies/{id_competencies_lesson_plan}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos da competência a serem atualizados.

Body Exemplo:

```
{
    "contentCompetenciesLessonPlan": "Desenvolver a capacidade de abstração e modelagem de problemas utilizando estruturas de dados avançadas."
}
```

Response (Status 200 - OK):

Retorno idêntico ao patch objetives, mas com o objeto id_competencies_lesson_plan atualizado.

4.8. patch themes

Atualiza um tema específico do plano de aula.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/lesson-plans/update/themes/{id_themes_lesson_plan}	Authorization: Bearer {{token}}, Content-Type:	Campos do tema a serem

		application/json	atualizados.

Body Exemplo:

```
{
   "titleThemesLessonPlan": "Introdução às listas duplamente encadeadas2"
}
```

Response (Status 200 - OK):

Retorno idêntico ao patch objetives, mas com o objeto id_themes_lesson_plan atualizado.

4.9. patch methodology

Atualiza uma metodologia de ensino específica do plano de aula.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/lesson-plans/update/methodology/{id_methodology_lesson_plan}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos da metodologia a serem atualizados.

Body Exemplo:

```
{
   "titleMethodologyLessonPlan": "Introdução às listas duplamente encadeadas"
}
```

Response (Status 200 - OK):

Retorno idêntico ao patch objetives, mas com o objeto id_methodology_lesson_plan atualizado.

4.10. patch topics

Atualiza um tópico principal dentro do plano de aula.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/lesson-plans/update/topics/{id_topics_lesson_plan}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos do tópico a serem atualizados.

Body Exemplo:

```
{
  "titleTopicsLessonPlan": "Introdução às Listas Encadeadas e suas Variações"
}
```

Response (Status 200 - OK):

Retorna o objeto do tópico atualizado, aninhado com o objeto pai do plano de aula.

```
{
  "message": "Plano de aula atualizado com sucesso.",
  "sucess": true,
  "updatedLessonPlan": {
      "id_topics_lesson_plan": "uuid",
      "titleTopicsLessonPlan": "string (updated)",
      // ... demais campos do tópico
      "lesson_plan": {
          "id_lesson_plan": "uuid",
          // ... campos principais do plano de aula pai
      }
  }
}
```

4.11. patch topics examples

Atualiza um exemplo específico dentro de um tópico.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/lesson-plans/update/examples-topics/{id_examples_topics}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos do exemplo a serem atualizados.

Body Exemplo:

```
{
  "titleExamplesTopicLessonPlan": "Editor de texto"
}
```

Response (Status 200 - OK):

Retorna o objeto do exemplo atualizado, aninhado com o tópico e o plano de aula pai.

```
{
   "message": "Plano de aula atualizado com sucesso.",
   "sucess": true,
   "updatedLessonPlan": {
       "id_examples_topics": "uuid",
       "titleExamplesTopicLessonPlan": "string (updated)",
       // ... demais campos do exemplo
       "topics_lesson_plan": { /* objeto do tópico pai */ }
   }
}
```

4.12. patch topics Activities

Atualiza uma atividade específica dentro de um tópico.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/lesson-plans/update/Activities-topics/{id_activities_topics}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos da atividade a serem atualizados.

Body Exemplo:

```
{
  "titleActivitiesTopicLessonPlan": "Discussão em Grupo"
}
```

Response (Status 200 - OK):

Retorno idêntico ao patch topics examples, mas com o objeto id_activities_topics atualizado.

4.13. patch topics connections

Atualiza uma conexão/relação específica dentro de um tópico.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/lesson-plans/update/connections-topics/{id_connections_topics}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos da conexão a serem atualizados.

Body Exemplo:

```
{
  "titleConnectionsTopics": "Editores de Texto"
}
```

Response (Status 200 - OK):

Retorno idêntico ao patch topics examples, mas com o objeto id_connections_topics atualizado.

4.14. patch inclusive-adaptation

Atualiza o bloco de adaptação inclusiva do plano de aula.

Método	URL	Headers	Body (JSON Structure)

PATCH	{{api-url}}/lesson-plans/update/inclusive-adaptation/{id_inclusive_adaptation_lesson_plan}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos a serem atualizados (visualImpairment, learningDifficulties, highAbilities).

Body Exemplo:

```
{
   "visualImpairment": "Disponibilizar o material didático em formato acessível (HTML, PDF com texto selecionável), utilizar fontes grandes e cores contrastantes nas apresentações, e fornecer audiodescrição das imagens e diagramas."
}
```

Response (Status 200 - OK):

Retorno idêntico ao patch objetives, mas com o objeto id_inclusive_adaptation_lesson_plan atualizado.

4.15. patch references

Atualiza uma referência específica do plano de aula.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/lesson-plans/update/references/{id_references_lesson_plan}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos da referência a serem atualizados.

Body Exemplo:

```
{
```

"contentReferencesLessonPlan": "Goodrich, M. T., Tamassia, R., & Goldwasser, M. H. (2014). Data Structures and Algorithms in Java (6th ed.). Wiley."
}

Response (Status 200 - OK):

Retorno idêntico ao patch objetives, mas com o objeto id_references_lesson_plan atualizado.

4.16. patch closure

Atualiza o bloco de encerramento do plano de aula.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/lesson-plans/update/closure/{id_closure_lesson_plan}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos do encerramento a serem atualizados (summary, reflection, nextSteps).

Body Exemplo:

{
"reflection": "Incentivar os alunos a refletir sobre a importância das estruturas de dados na resolução de problemas de programação e sobre as vantagens e desvantagens das listas duplamente encadeadas em relação a outras estruturas."
}

Response (Status 200 - OK):

Retorno idêntico ao patch objetives, mas com o objeto id_closure_lesson_plan atualizado.

4.17. patch homework

Atualiza o bloco de lição de casa do plano de aula.

Método	URL	Headers	Body (JSON Structure)
PATCH	{{api-url}}/lesson-plans/update/homeworks/{id_homework_lesson_plan}	Authorization: Bearer {{token}}, Content-Type: application/json	Campos da lição de casa a serem atualizados (description, objective).

Body Exemplo:

```
{
    "description": "Implementar uma lista duplamente encadeada genérica em Java que possa armazenar diferentes tipos de dados. Escrever testes unitários abrangentes para verificar a correção de todos os métodos da lista."
}
```

Response (Status 200 - OK):

Retorno idêntico ao patch objetives, mas com o objeto id_homework_lesson_plan atualizado.

5. gemini

Endpoints para criação de conteúdo usando IA (Gemini).

5.1. create lesson

Cria um plano de aula usando a IA.

Método	URL	Headers	Body (JSON Structure)

POST	{{api-url}}/gemini/create-lesson-plan	Authorization: Bearer {{token}}, Content-Type: application/json	Input: Parâmetros para a geração do plano de aula (tópicos, objetivos, nível).

Body Exemplo:

```
{
  "subjectLessonPlan": "Estrutura de Dados em Java",
  "descriptionLessonPlan": "Aula sobre o funcionamento, implementação e manipulação de listas duplamente encadeadas em Java.",
  "gradeLevelLessonPlan": "Ensino Superior",
  "complexityLevelLessonPlan": "Avançado",
  "durationMinutesLessonPlan": 120,
  "objectives": [ /* array de objetivos */ ],
  "teachingMethodologies": [ /* array de metodologias */ ],
  "themes": [ /* array de temas */ ]
}
```

Response (Status 200 - OK):

Retorna o objeto completo do Plano de Aula gerado pela IA, com a mesma estrutura aninhada de sub-objetos do endpoint add lesson_plan (exceto pelos IDs e timestamps, que seriam adicionados no próximo passo, que é salvar o plano no sistema).

```
{
   "subjectLessonPlan": "string",
   "descriptionLessonPlan": "string",
   "gradeLevelLessonPlan": "string",
   "complexityLevelLessonPlan": "string",
   "durationMinutesLessonPlan": "number",
   "generalObjective": "string",
   "specificObjectives": [ /* array de objetos */ ],
   "competencies": [ /* array de objetos */ ],
   "themes": [ /* array de objetos */ ],
   "teachingMethodologies": [ /* array de objetos */ ],
   "topics": [ /* array complexo de objetos de tópicos com exemplos, atividades e conexões */ ],
   "homework": { /* objeto de casa */ },
   "inclusiveAdaptation": { /* objeto de adaptação */ },
   "references": [ /* array de objetos */ ],
   "closure": { /* objeto de encerramento */ }
}
```

5.2. create exercises

Cria exercícios usando a IA.

Método	URL	Headers	Body (JSON Structure)
POST	{{api-url}}/gemini/create-exercises	Authorization: Bearer {{token}}, Content-Type: application/json	Input: Parâmetros para a geração de exercícios (tópico, nível, tipos de exercícios).

Body Exemplo:

```
{
  "subject_exercises": "Parasitologia",
  "description_exercises": "Exercícios para revisão de ectoparasitas...",
  "grade_level_exercises": "Ensino Superior",
  "complexity_level_exercises": "Avançado",
  "duration_minutes_exercises": 90,
  "objectives_exercises": [ /* array de objetivos */ ],
  "themes_exercises": [ /* array de temas */ ],
  "exercises": [
    { "type_exercise": "multipla-escolha", "bloom_level": "analisar" },
    { "type_exercise": "discursiva", "bloom_level": "avaliar" },
    { "type_exercise": "verdadeiro-falso", "bloom_level": "lembrar" }
  ]
}
```

Response (Status 200 - OK):

Retorna o objeto completo do Exercício gerado pela IA, com a mesma estrutura aninhada de sub-objetos do endpoint add exercise (exceto pelos IDs e timestamps, que seriam adicionados ao salvar no sistema).

```
{
    "subject_exercises": "string",
    "description_exercises": "string",
    "grade_level_exercises": "string",
    "complexity_level_exercises": "string",
    "duration_minutes_exercises": "number",
    "objectives_exercises": [ /* array de objetos */ ],
    "themes_exercises": [ /* array de objetos */ ],
    "exercises": [
```

```
        {
            "type_exercise": "string",
            "title_exercise": "string",
            "content_exercise": "string",
            "options_exercise_multipla_escolha": [ /* array de objetos de opções */ ],
            "correct_answer_exercise": "string",
            "explanation_exercise": "string",
            "bloom_level": "string"
        }
        // ... more exercise items
    ]
}
```