

# API Aprenda S

O AprendaS é um aplicativo educacional com a proposta inovadora de transformar o aprendizado em uma experiência envolvente e recompensadora. Com um ambiente simples e intuitivo, o AprendaS introduz um sistema de gamificação em que os alunos ganham pontos ao responder perguntas relacionadas aos temas estudados em sala de aula. Esses pontos os ajudam a subir em um ranking estadual, onde aqueles que obtêm mais pontos no final do mês são premiados com descontos nas mensalidades e brindes exclusivos.

## Requisitos

---

- [Node.js](#) instalado

## Passo a Passo para Executar a Aplicação

---

1. **Clone o repositório ou copie os arquivos para um diretório local.**
2. **Instale as dependências do projeto:**

Abra o terminal na pasta onde está o projeto e execute o comando:

```
npm init -y  
npm install express fs
```

3. **Caso o npm não esteja sendo reconhecido, execute como administrador o comando:**

- Set-ExecutionPolicy RemoteSigned

4. **Inicie o servidor:** node server.js

5. **A API estará rodando em:**

`http://localhost:3000`

## Rotas da API

# Usuario

## 1. Login (POST /login)

Realiza o login do usuário com base em email e senha.

### body

```
{  
  "email": "novo.aluno@example.com",  
  "senha": "senha123"  
}
```

json

## 2. Listar alunos (GET /alunos)

Lista todos os alunos cadastrados

Exemplo GET:

```
http://localhost:3000/alunos
```

## 3. Adicionar Novo Usuário (POST /alunos)

Adiciona um novo usuário ao banco de dados.

### body

```
{  
  "nome_completo": "Novo Aluno",  
  "cidade": "Curitiba",  
  "curso": "Técnico em Desenvolvimento de Sistemas",  
  "email": "novo.aluno@example.com",  
  "senha": "senha123"  
}
```

json

## 4. Atualizar Informações do Usuário (PUT /alunos/:id)

Atualiza os dados de um aluno já cadastrado. Parâmetros da URL: id: ID do aluno.

```
{  
  "nome_completo": "Novo nome",  
  "email": "novo.aluno@example.com",  
  "senha": "senha123"  
}
```

json

```
"cidade": "Nova Cidade"  
}
```

## DISCIPLINAS

---

### 5. Todos os Cursos (GET /cursos)

Retorna todos os cursos cadastrados.

Exemplo GET:

```
http://localhost:3000/cursos
```

### 6. Disciplinas do Curso do Usuário (GET /disciplinas-usuario/:id)

Retorna as disciplinas do curso associado ao aluno logado. **Parâmetros da URL:** id: ID do aluno.

Exemplo GET:

```
http://localhost:3000/disciplinas-usuario/1
```

### 7. Todas as Disciplinas (GET /disciplinas)

Retorna todas as disciplinas de todos os cursos.

Exemplo GET:

```
http://localhost:3000/disciplinas
```

## RANKING

---

### 8. Todos os Dados de Ranking (GET /ranking)

Retorna todos os dados do ranking.

Exemplo GET:

`http://localhost:3000/ranking`

## 9. Ranking dos Alunos do Mesmo Curso (GET /ranking-curso/:id)

Retorna os dados de ranking referentes aos alunos do mesmo curso do aluno logado. Parâmetros da URL: id: ID do aluno.

Exemplo GET:

`http://localhost:3000/ranking-curso/1`

## 10. Top 10 Ranking do Mesmo Curso (GET /ranking-top10/:id)

Retorna os 10 primeiros colocados do ranking do curso do aluno logado. Parâmetros da URL: id: ID do aluno.

Exemplo GET:

`http://localhost:3000/ranking-top10/1`