**UniForum**

Este documento describe cada microservicio del proyecto **UniForum**, su funcionalidad y un ejemplo de cómo usarlo desde **Postman**.

**Dominio: Gestión de Usuarios**

**1. registro-usuarios (Spring Boot)**

**Función:** Registra estudiantes y universidades.  
**POST /students**

json

CopiarEditar

{

"idStudent": "stu001",

"firstName": "Nombre",

"lastName": "Vera",

"email": "nombre@example.com",

"password": "clave123",

"university": {

"idUniversity": "001"

}

}

**GET /students** - Lista todos los estudiantes.

**2. auth-service (Spring Boot)**

**Función:** Login de usuarios con JWT.  
**POST /auth/login**

json

CopiarEditar

{

"email": "nombre@example.com",

"password": "clave123"

}

**Respuesta:**

json

CopiarEditar

{

"token": "eyJhbGciOiJIUzI1NiIsInR..."

}

**3. authorization-service (Spring Boot)**

**Función:** Valida JWT y extrae el rol del usuario.

Requiere header: Authorization: Bearer <token>

**4. perfil-service (FastAPI)**

**GET /perfil/me**

Header: Authorization: Bearer <token>

**5. notification-service (FastAPI)**

**Función:** Escucha eventos (RabbitMQ) y envía correos.

**6. password-service (FastAPI)**

**POST /password/recover**

json

CopiarEditar

{

"email": "nombre@example.com"

}

**💬 Dominio: Foro Académico**

**7. discussion-service**

**POST /posts**

json

CopiarEditar

{

"id\_post": "p001",

"title": "Tema 1",

"content": "Contenido",

"student\_id": "stu001"

}

**8. reply-service**

**POST /comments**

json

CopiarEditar

{

"id\_comment": "c001",

"content": "Respuesta",

"post\_id": "p001",

"student\_id": "stu001"

}

**9. moderation-service**

**GET /moderation/comments**

**10. voting-service**

**POST /votes**

json

CopiarEditar

{

"id\_vote": "v001",

"student\_id": "stu001",

"post\_id": "p001",

"vote\_type": "upvote"

}

**11. tagging-service**

**POST /tags**

json

CopiarEditar

{

"id\_tag": "t001",

"post\_id": "p001",

"name": "python"

}

**12. history-service**

**POST /history**

json

CopiarEditar

{

"id\_history": "h001",

"post\_id": "p001",

"previous\_content": "Antes",

"new\_content": "Después"

}

**🎓 Dominio: Gestión de Cursos**

**13. course-service**

**POST /courses**

json

CopiarEditar

{

"id\_course": "c001",

"name": "Base de Datos",

"description": "Intro curso"

}

**14. enrollment-service**

**POST /enrollments**

json

CopiarEditar

{

"id\_enrollment": "e001",

"student\_id": "stu001",

"course\_id": "c001"

}

**15. course-forum-link-service (Node.js)**

**POST /links**

json

CopiarEditar

{

"course\_id": "c001",

"post\_id": "p001"

}