/api/auth/login

- Método: POST.
- Ruta: `/api/auth/login`.
- Autenticación: no requiere (es el punto de entrada).
- Controlador: `AuthController.login` → Servicio: `AuthService.iniciarSesion`.

Request

- Headers: `Content-Type: application/json`.
- Body:
    - `email`: string en formato válido.
    - `password`: string.

Validaciones y reglas

- VO `Email` valida formato y normaliza a minúsculas.
- Busca usuario por email; si no existe o la contraseña no coincide con el hash bcrypt, lanza `AuthError`.
- Genera JWT firmado con campos `sub`, `email`, `rol`; expiración default 1h (`JwtTokenProvider`).

Respuestas

- 200 `{ "token": "<jwt>" }`.
- 401 por credenciales inválidas.
- 400 si el email no pasa la validación de formato.

Consideraciones

- El rol en el token se utiliza por los middlewares para autorizar rutas protegidas (`register`, reclamo y atención).