

Delivery #1: Discovery & Reverse Engineering

TASK #3 Backlog Recovery:

Generación de una lista con las historias de usuario basado en los bounded context encontrados con anterioridad.

Usaremos el formato clásico:

Como ...
Quiero ...
Para ...

y debajo indicaremos la trazabilidad técnica real de cada uno.

HU-USER-01 – Registro de usuario

Historia

Como visitante
Quiero registrarme con email y contraseña
Para poder autenticarse y usar la plataforma

Trazabilidad técnica

- `conduit/apps/users/models.py` → modelo User
- `conduit/apps/users/serializers.py` → UserRegistrationSerializer
- `conduit/apps/users/views.py` → RegistrationAPIView
- `conduit/apps/users/urls.py` → endpoint `/api/users`

HU-USER-02 – Autenticación de usuario

Historia

Como usuario registrado
Quiero iniciar sesión y obtener un token
Para acceder a funcionalidades protegidas

Trazabilidad técnica

- `conduit/apps/users/serializers.py` → LoginSerializer
- `conduit/apps/users/views.py` → LoginAPIView
- `conduit/apps/users/authentication.py` → JWT
- `conduit/apps/users/urls.py` → endpoint `/api/users/login`

HU-PROFILE-01 – Ver perfil de usuario

Historia

Como usuario autenticado
Quiero ver el perfil público de otro usuario
Para conocer su información y si lo sigo

Trazabilidad técnica

- `conduit/apps/profiles/models.py` → `Profile`
- `conduit/apps/profiles/serializers.py` → `ProfileSerializer`
- `conduit/apps/profiles/views.py` → `ProfileRetrieveAPIView`
- `conduit/apps/profiles/urls.py` → `/api/profiles/{username}`

HU-PROFILE-02 – Seguir a un usuario

Historia

Como usuario autenticado
Quiero seguir a otro usuario
Para ver su contenido en mi feed

Trazabilidad técnica

- `conduit/apps/profiles/models.py` → relación `following`
- `conduit/apps/profiles/views.py` → `ProfileFollowAPIView`
- `conduit/apps/profiles/urls.py` → `/api/profiles/{username}/follow`

HU-ARTICLE-01 – Publicar un artículo

Historia

Como usuario autenticado
Quiero publicar un artículo
Para compartir contenido con otros usuarios

Trazabilidad técnica

- `conduit/apps/articles/models.py` → `Article`
- `conduit/apps/articles/serializers.py` → `ArticleSerializer`
- `conduit/apps/articles/views.py` → `ArticleCreateAPIView`
- `conduit/apps/articles/urls.py` → `/api/articles`

HU-ARTICLE-02 – Marcar artículo como favorito

Historia

Como usuario autenticado
Quiero marcar un artículo como favorito
Para guardarlo y expresar interés

Trazabilidad técnica

- `conduit/apps/articles/models.py` → relación favorites
- `conduit/apps/articles/serializers.py`
- `conduit/apps/articles/views.py` → `ArticleFavoriteAPIView`
- `conduit/apps/articles/urls.py` → `/api/articles/{slug}/favorite`

HU-COMMENT-01 – Comentar un artículo

Historia

Como usuario autenticado
Quiero agregar un comentario a un artículo
Para participar en la conversación

Trazabilidad técnica

- `conduit/apps/comments/models.py` → `Comment`
- `conduit/apps/comments/serializers.py`
- `conduit/apps/comments/views.py` → `CommentCreateAPIView`
- `conduit/apps/comments/urls.py` → `/api/articles/{slug}/comments`

HU-COMMENT-02 – Eliminar comentario propio

Historia

Como autor de un comentario
Quiero eliminar mi comentario
Para mantener control sobre mi contenido

Trazabilidad técnica

- `c`
- `conduit/apps/comments/views.py` → `CommentDestroyAPIView`
- `conduit/apps/comments/models.py`
- Validación de autoría en la vista

HU-TAG-01 – Listar tags disponibles

Historia

Como usuario
Quiero ver la lista de tags
Para filtrar y explorar artículos

Trazabilidad técnica

- `conduit/apps/tags/models.py` → Tag
- `conduit/apps/tags/serializers.py`
- `conduit/apps/tags/views.py` → TagListAPIView
- `conduit/apps/tags/urls.py` → `/api/tags`

HU-TAG-02 – Asociar tags a un artículo

Historia

Como autor de un artículo
Quiero asignar tags al publicar
Para clasificar mejor mi contenido

Trazabilidad técnica

- `conduit/apps/articles/models.py` → relación con Tag
- `conduit/apps/articles/serializers.py`
- Lógica en creación/actualización de artículos.