En el presente docuemento se detalla la implementacion los componentes intervinentes en el workflow de autenticacion.

La autenticacion esta basada en JWT <https://jwt.io/introduction/> donde se define una forma compacta autocontenida para transmitir la informacion entre clienet y servidor (recurso y/o de seguridad). Esta informacion esta digitalmente firmada utilizando HMAC , RSA o **ECDSA**



**Resource Owner:**

The entity or person (user) that owns the protected **Resource**.

**Resource Server:**

Es el servidor que contiene las api que permiten el acceso directo a los recursos portegidos.

**Client Applications:**

Es cualquier app que (software) solicite acceso a algun recurso protegido en el Resource Server (auidience)

Ver Registro de Client Applications en Resource Owners

## Authorization Server (Issuer)

Es el server que genera y verifica las credenciales del usuario, genera el Token y lo retorna al cliente en caso de exito. Para ello requiere chequar el registro del Resourse Owner

Nombre físico fuente fwk**Authentication**WebApi

APIS

### Grant Tocken - Conceder un JWT

POST <https://localhost:51000/oauth/token> Este metodo es el principal y es el que genera el token

Host: localhost: 51000

Content-Type: application/x-www-form-urlencoded

**Parametros**:

* **username** . nombre de uysuario con el que esta registado en la bd de negocio
* **password**: password del usuario
* **grant\_type** : password
* **client\_id**: Nombre o identificador de la aplicacion en cuestion que utilizara el Issuer para firmar el jwt: Recordar que el Issuer puede ser generador de tokens para mas de una app.
* **client\_secret**: clave secreta con la que se gfenero el Hash que

Nota: Para que un client\_id sea valido este debera ser previamente registrado y accesible al **Authorization Server**

### Registro de Client Applications en Resource Owners

Las apps clientes se almacenan en una tabla propuesta por el bloque de seguridad Fwk.Security.Identity llamada SecurityClients (en otro apartado se muestra como instalar las tablas de este bloque)

Para efectuar el registro se debe llamar al siguiente Endpoint:

POST /api/account/createApplication HTTP/1.1

Host: localhost: 51000

Content-Type: application/json

{

"**Id**":"webChatCustomer", //Identificador de la aplicacion

"**Secret**":"socratesSecret", //

"**Name**":"CustommersChat", //Nombre descriptivo de la aplicacion

"ApplicationType":"1",

"Active":true,

"RefreshTokenLifeTime":20, //tiempo de expiracion del token jwt generado

"AllowedOrigin":1

}

Por lo tanto llamando a esta api se permite crear una aplicacion en SecurityClients Donde SecurityClients es un repositorio accesible a travez del bloque Fwk.Security.Identity para registrar todas las App que requieran autenticacion y autorizacion de usuarios

### Refresh Token