**Main Roles**

* **User** – the person who uses the application. The users usually work with a **frontend application,** which we call a client.
* **Client** – application that calls a backend and needs authentication and authorization. The client can be a web app, mobile app, desktop app or a separate backend service.
* **Resources Server** – backend application that authorizes and serves calls sent by one or more client applications.
* **Authorization Server** – application that implements authentication and safe storage of credentials

**A diagram of a person and a computer

Description automatically generated**

**Main Steps**

1. The user tries to use the client application to execute a particular use case.
2. The client application knows it can’t call its backend without first having a token that will allow it to get authorized. The client requests such an access token from the authorization server.
3. Following the client app’s request, the authorization server issues a token and sends it to the client app.
4. The client uses the token to send requests to its backend (the resource server).
5. The resource server authorizes the client’s request. If authorized successfully, the resource server executes the client’s request and replies back.
6. The client shows the result to the user

A diagram of a computer network

Description automatically generated

After logging in and receiving an authorization code, the client must make an additional request to acquire the access token. During this request, the client is required to verify its identity using its credentials. This method increases the challenge for anyone attempting to illicitly obtain the access token, as they would need to intercept the authorization code and also know the client’s credentials

A diagram of a computer system

Description automatically generated