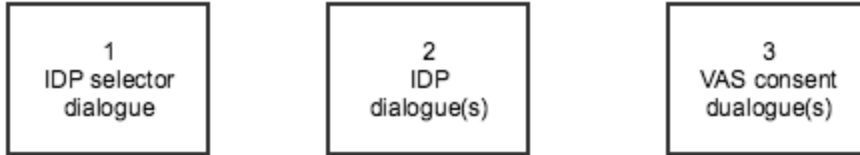


# User experience

The OIDC Provider from BankID requires user interaction at three stages in its flow as illustrated below:

1. Selection of IDP-service by the end-user if the OIDC Client has not already requested a particular IDP via the [login\\_hint](#) parameter
2. Entry of required userID and credentials for the designated IDP.
3. Display and accept of consent for the requested VAS service(s).



The dialogues shown above corresponds to the default user-experience that comes with the OIDC Provider for the combination of BankID as IDP-service and TINFO as VAS-service. The OIDC Provider offers a default user-experience also for other combinations of IDP and VAS. Note that restrictions may apply on how the OIDC Provider can implement a default user-experience for or stage 2 and/or 3 depending on the security policies for the IDP and/or VAS (s) in question.

This release of the OIDC Provider supports web-based OIDC Client applications using browser re-direction to govern user-experience. This includes support for app-based OIDC Client applications using web-views. The OIDC Client is in either case in control on how and where each of the involved dialogues is integrated with the OIDC Client application. Three common user-experience alternatives are as follows:

- **Re-direct:** Re-direct of the entire application page
- **Inline:** Re-direct of an iframe embedded in the application (mother) page
- **Window:** Re-direct of a pop-up (window) from the application (mother) page

The [JS Connector](#) may simplify the task to integrate with the OIDC Provider for front-end based applications. Among other things, it supports each of the user-experience alternatives from the above list.

This release of the OIDC Provider also supports hybrid scenarios in which the OIDC Client application itself is web-based whereas the involved IDP and/or VAS are app-based. This is referred to as a decoupled user-experience



This release of the OIDC Provider does not support pure app-based applications using a completely embedded (API-based) user-experience. A future release may include such support either via deep-linking between the OIDC Client app and a designated OIDC Provider App, or via integration of a OIDC Provider SDK into the OIDC Client app.

For re-direct scenarios the OIDC Clients may in addition replace the default dialogues with customized releases for each of the 3 stages. See [Customization of user-experience](#) for further details. Note however that restrictions may apply on how the OIDC Client can implement custom user-experience for stage 2 and/or 3 depending on the security policy for the IDP and/or VAS(s) in question.