User Authentication & Registration

[Design System Image]

Most applications require some form of user authentication to keep unauthorized users out, and a way to add authorized users to the application.

[DemoCard Component - Interactive React component]

[TOC Component - Interactive React component]

Variations

User authentication contains four primary workflows:

- Registration
- Joining an organization
- Login
- Forgot / Reset password

All of these workflows should follow the rules specified by our forms and validation pattern, and multi-step workflows should be connected using our steppers pattern.

[MaterialDesignDescription Component - Interactive React component]

} />

[MaterialDesignDescription Component - Interactive React component]

} />

Your authentication and registration workflows should work equally well on different screen sizes. On mobile, these should be presented as full-screen layouts, whereas on desktop, they should be presented in a card centered on the screen as an overlay.

Registration

Registration is the process of user account creation. Typically, users can either register themselves (self-registration) or you can invite them to join your application (invitation-based).

Self-Registration

If you do not need to restrict who can register for your application, you can allow users to self-register. The self-registration workflow includes the following steps:

- Accept EULA
- Enter email address
- Receive an email for verification
- Verify email
- Create password
- Add account details
- Success: account created

Registration via Invitation

If you need to restrict who is able to use your application, you may choose to go with a invitation-based registration. In this situation, a system admin sends an email to authorized users inviting them to register. The invitation-based registration workflow includes the following steps:

- Receive an email for invitation
- Accept EULA
- Create password
- Add account details
- Success: account created

Screens

The majority of the registration screens are the same between invitation-based and self-registration. The primary difference is in how the registration workflow is initiated, and whether the email needs to be verified. Use the following screens for the registration workflow where appropriate.

[Design System Image]

> *The purpose of a success screen is to assure the user that something has been achieved. These screens can be an opportunity to perform more product branding — you are encouraged to use animations and graphics on these screens.*

[Design System Image]

Joining an Organization

Some applications may require users to join specific groups or organizations after creating an account. Generally, there are three ways for a user to join an organization:

Request Access
This occurs when a user has an account but need to access an invitation-only organization. This workflow involves the following steps:

- Enter organization identifier
- Success: request sent
- Receive an email invitation
- Success: organization joined

Join with Invitation
Joining via invitation occurs when a system admin extends an invitation for a user to join a group. If the user does not yet have an account, they will first proceed through the invitation-based registration workflow followed by the screens for joining the organization. This workflow includes the following steps:

- Receive an email invitation
- Success: organization joined

Screens

The majority of the joining-an-organization screens are the same regardless of the underlying mechanism. Use the following screens for the joining-an-organization workflow where appropriate.

[Design System Image]

[Design System Image]

Login

Logging in can usually be accomplished in one of two ways: basic login or login via a third-party.

Basic Login
A simple login presents fields for the users to enter their credentials and a button to submit them for verification.

The login screen should always present information on how to create an account and what to do if a user has forgotten their credentials — these buttons should trigger the Registration and Forgot Password (see below) workflows .

[Design System Image]

[Design System Image]

Login via Third Party

Some applications, especially consumer-facing ones, can be authorized via third-party accounts / credentials. Different platforms may have different requirements for this. Here are some commonly used platforms and their instructions:

- Login with Amazon
- Google Sign-in
- Sign in with Apple
- Facebook Login
- Sign in with Microsoft

Forgot / Change Password

Forgot password

The login screen for an application should also present users with a way to reset their password if they have forgotten it. This is typically displayed as a hyperlink below the credential fields. Clicking the link will present users with a field to enter their e-mail address. They will then receive an email with a hyperlink to reset their password. This workflow involves the following steps:

- Enter email address
- Success: reset instructions sent
- Receive an email for resetting the password
- Create a new password
- Success: password updated

[Design System Image]

[Design System Image]

Change Password
When the users are logged in, they have the option to change their password by going through the change password workflow, typically triggered from within the user settings. This workflow includes the following steps:

- Change password
- Success: password updated

> *After a successful password change, users should be logged out and forced to log in with the new credentials.*

[Design System Image]

[Design System Image]

Behaviors

Email Communications

Email communications are frequently used throughout the user authentication workflow to verify the user's identity. Users receive emails for invitations, address verification, confirmation codes, etc.

Emails are also sent to confirm security-related changes to a user's account, such as changing a password, updating a profile, etc. You should never allow users to unsubscribe from security-related email notifications.

[Design System Image]

Terminated Workflows & Expired Sessions

During the registration, if a user leaves the application before creating a password for their account, the whole session is treated as expired, and upon returning, they must start the registration process over again.

If an invitation link or verification code has expired, the users should be able to request for another invitation link or verification code.

[Design System Image]

Security Concerns

Error Messages

When a user encounters an error — such as a password mismatch or the server is temporarily down — they should be notified with an appropriate error message. However, be very careful when presenting error messages to the users. You should only show what is necessary to recover from the problem, such as "incorrect email or password" or "server unavailable" — do not expose information that could compromise the security of your application.

> *Do **NOT** provide messages like "User does not exist," "Incorrect Password," etc.*

[Design System Image]

Exposing Credentials

Unlike a phone application, a shared HMI application is more vulnerable to outside threats via showing credentials. These projects should **NOT** implement password visibility toggles or "remember me" functionality.

Two-Factor Authentication

The examples on this page used email as the login credentials. However, if your project requires additional security, consider using two-factor authentication. Two-factor authentication involves sending an additional confirmation code (typically via email or SMS) to verify a user's identity after they enter their password.

Developers

We recommend using the following packages to construct these workflows. Additionally, you should familiarize yourself with the components used for the Forms and Overlays patterns.

Angular:
- @brightlayer-ui/angular-auth-workflow

React:
- @brightlayer-ui/react-auth-workflow

React Native:
- @brightlayer-ui/react-native-auth-workflow