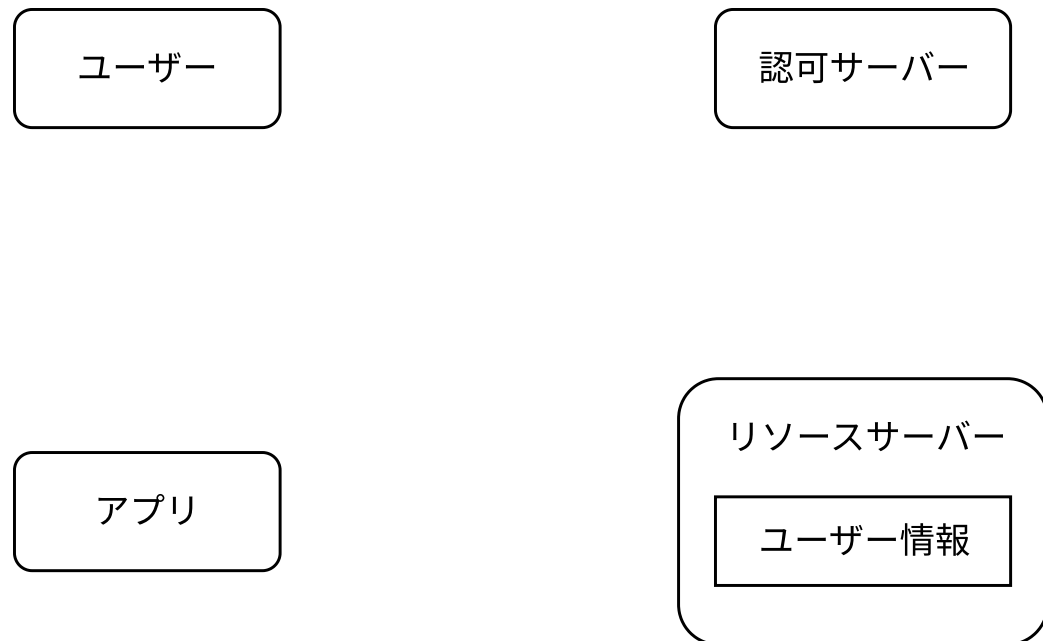


OAuth 2.0 を利用したアプリケーションの作成

Oauth 2.0とは



ユーザー（リソースオーナー）がサードパーティのアプリケーションにユーザー情報（リソース）へのアクセスを許可するためのフレームワーク

<https://www.rfc-editor.org/rfc/rfc6749>

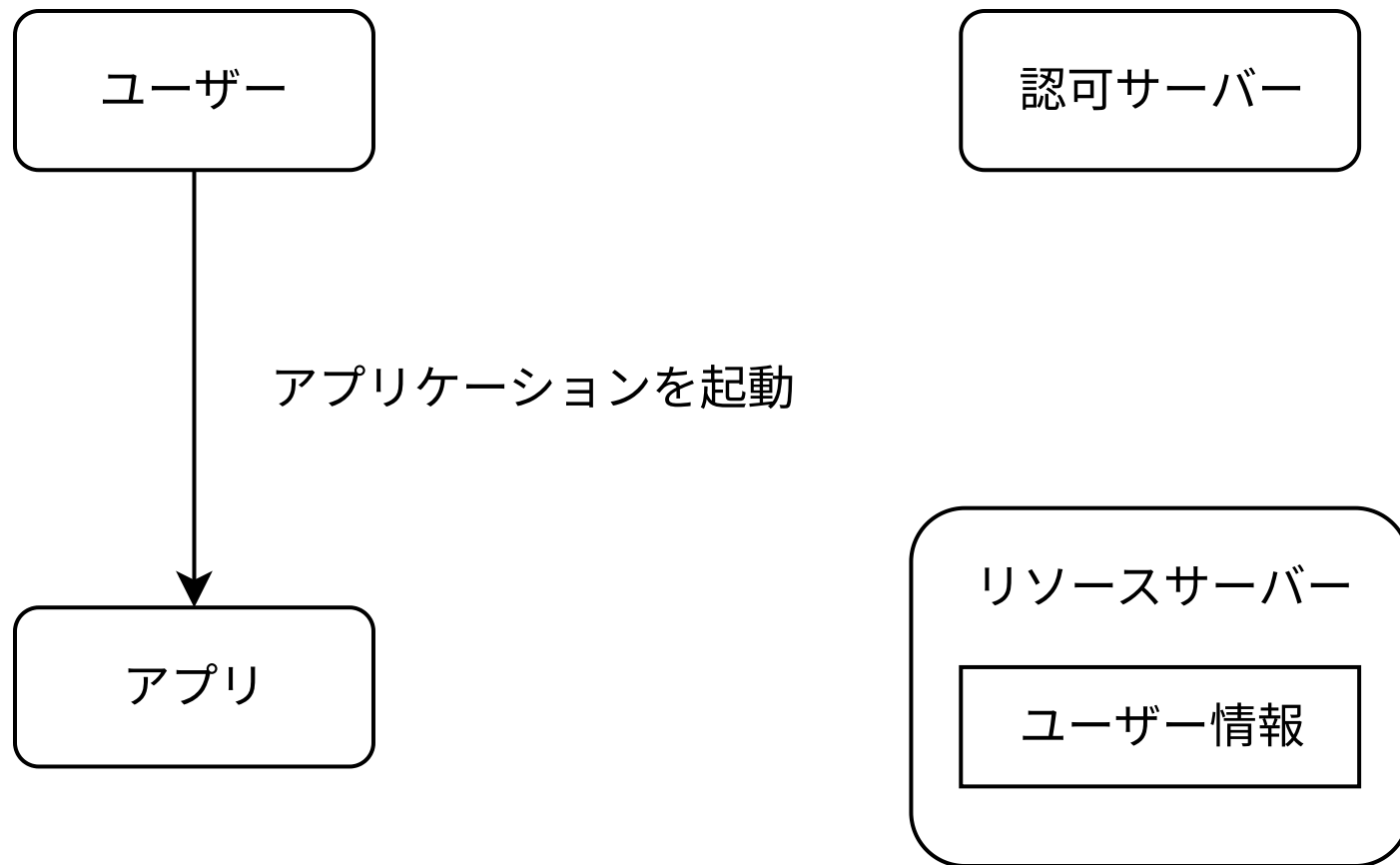
作成したアプリ概要

OAuth 2.0の認可コードフローを利用して、アプリケーションがユーザーの名前とメールアドレスを取得する

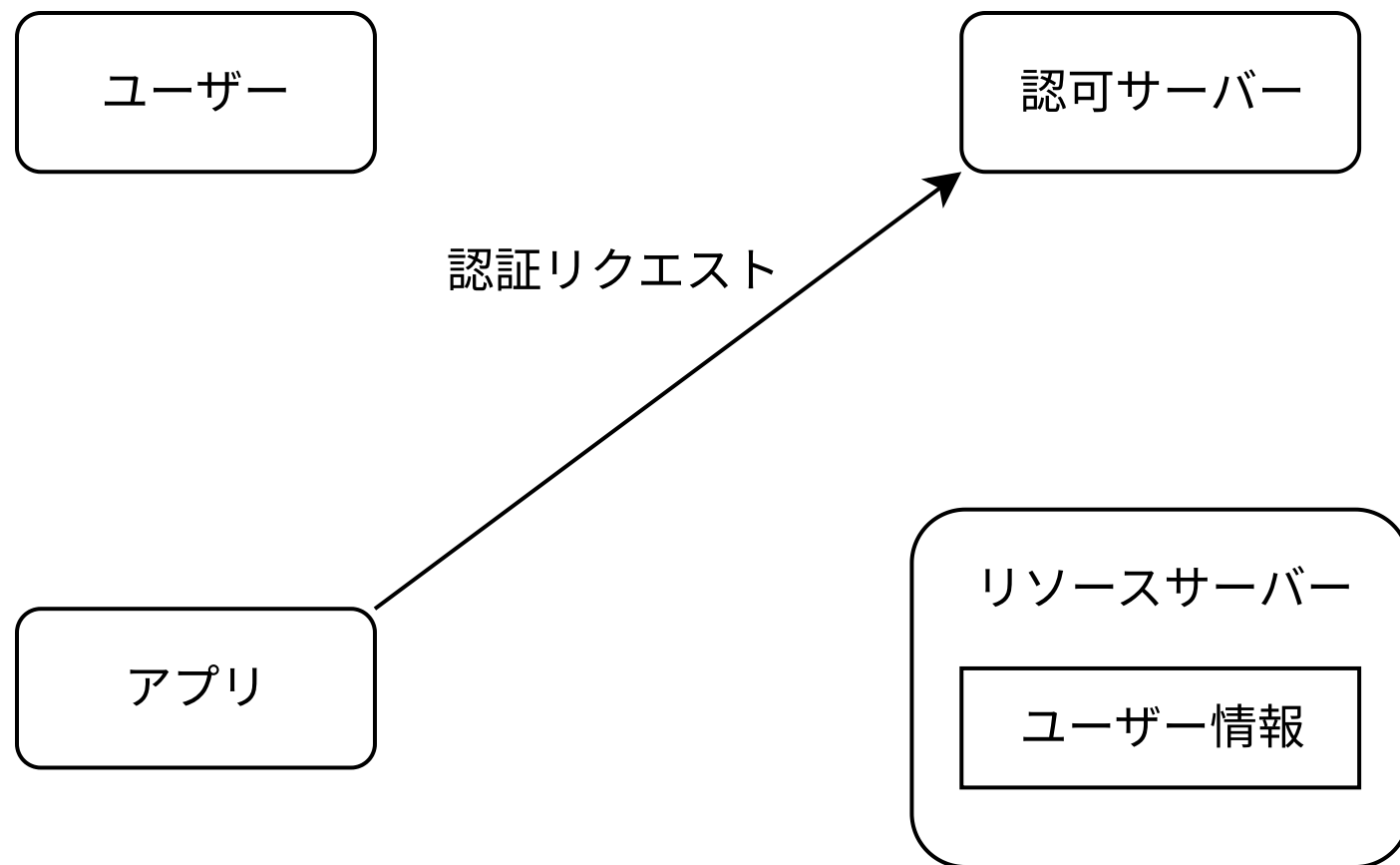
```
.../oauth-test/oauth-client main ! 11:31
> node index.js auth
OAuth2 client running on http://localhost:9010
ブラウザで http://localhost:9010 にアクセスして認証を開始してください
認証成功！トークンを .secretファイルに保存しました

.../oauth-test/oauth-client main ! 11:31
> node index.js get-name
ユーザー名: foo
メールアドレス: test@example.com
```

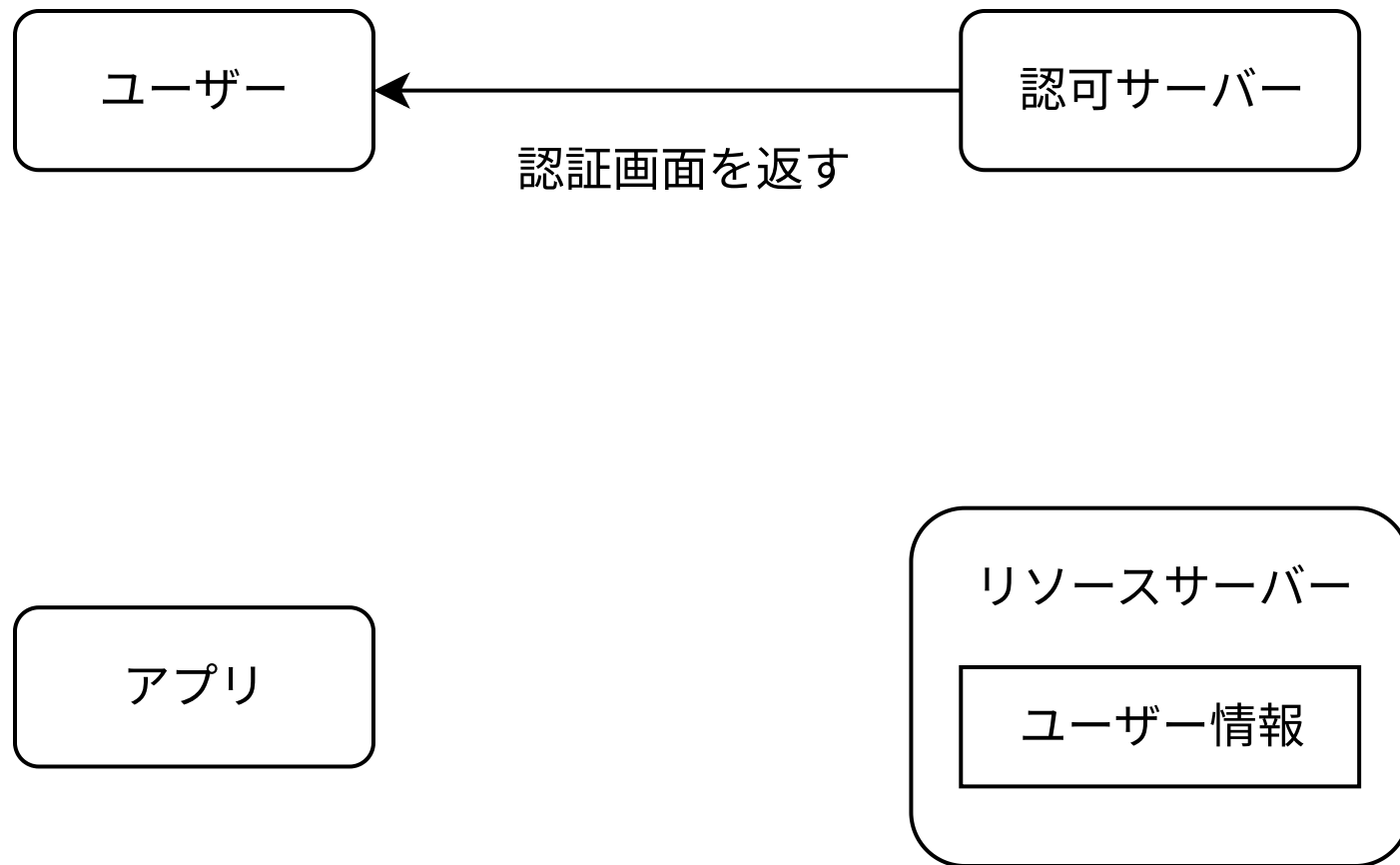
認証、認可フロー（認可コードフロー）



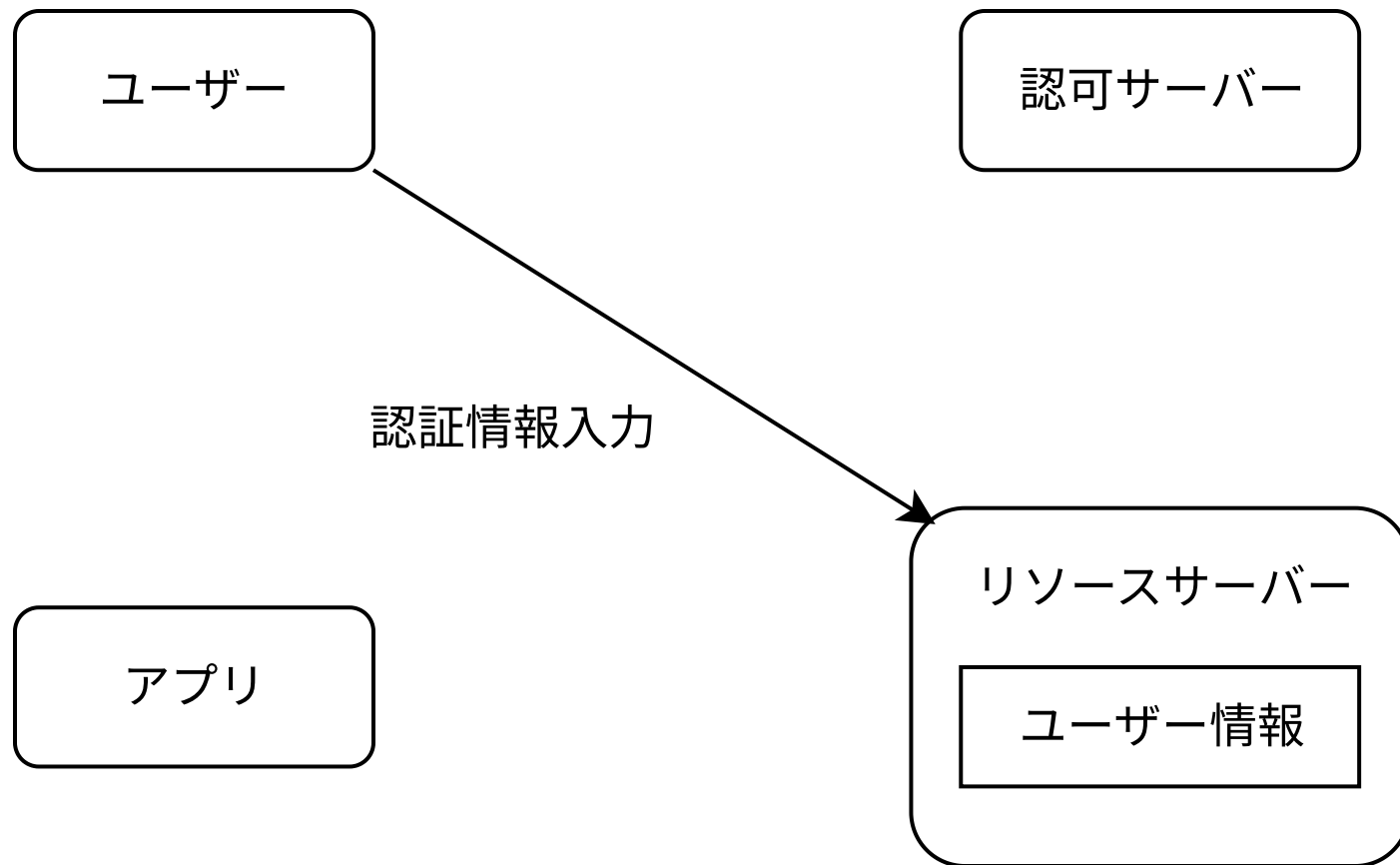
認証、認可フロー



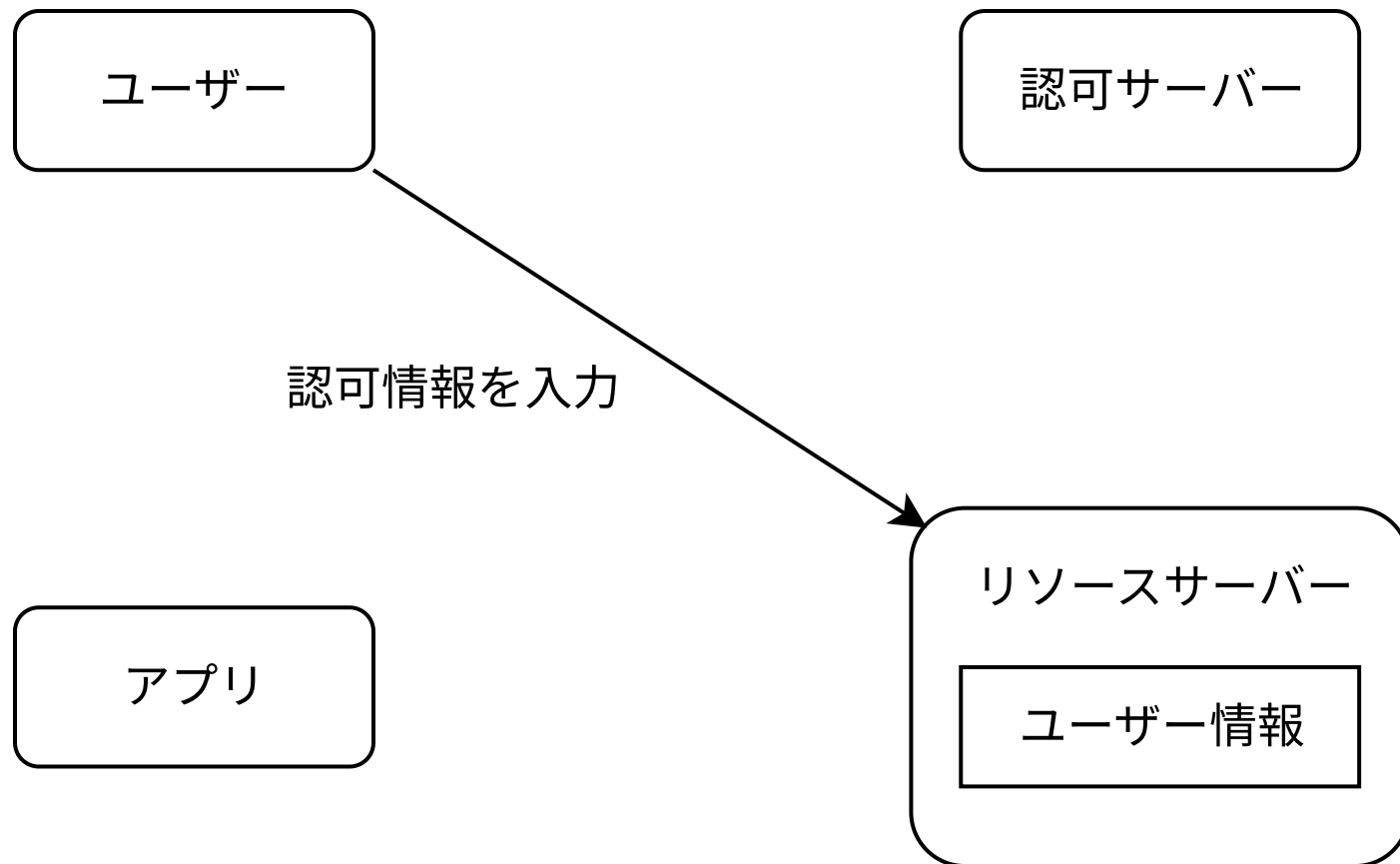
認証、認可フロー



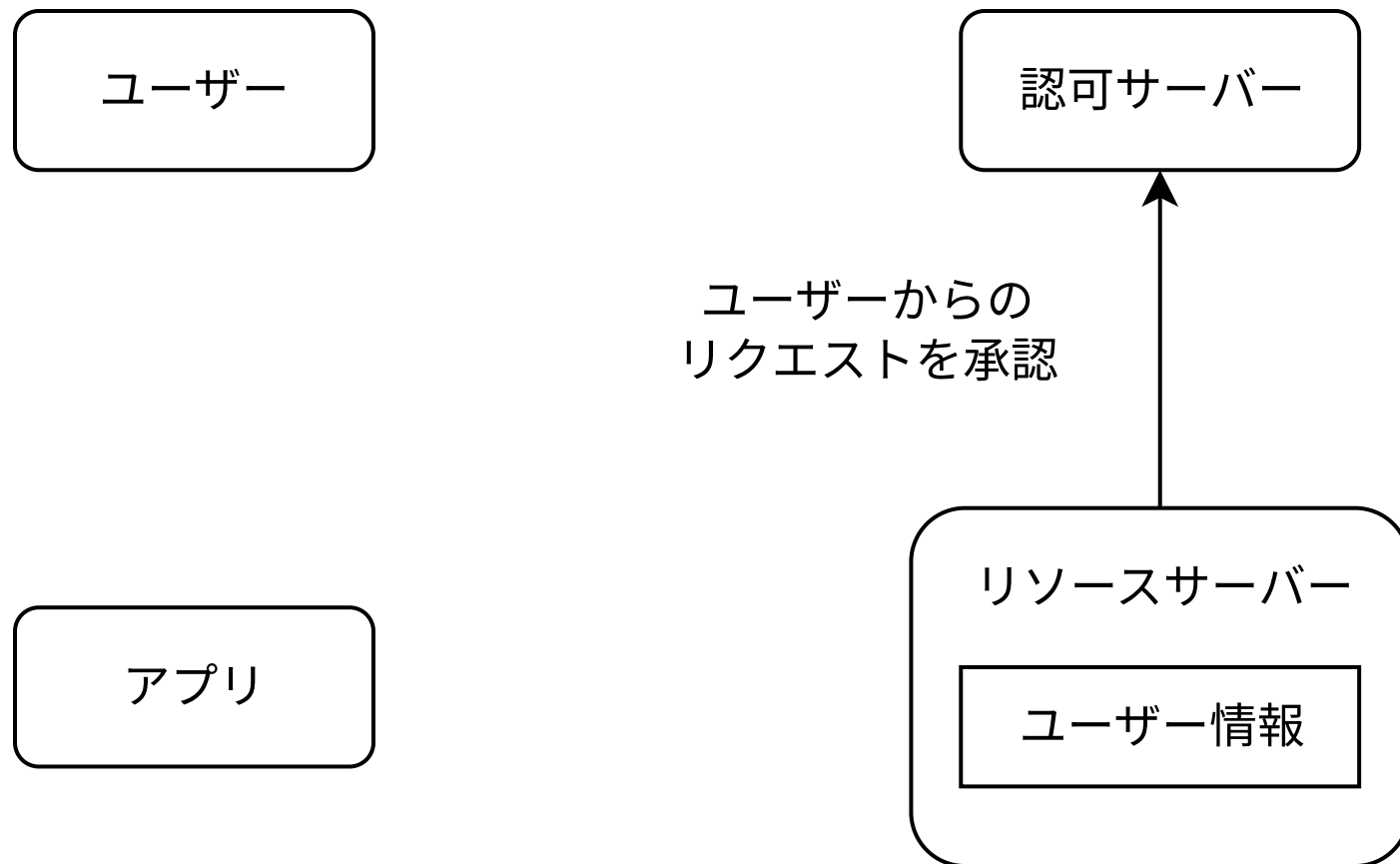
認証、認可フロー



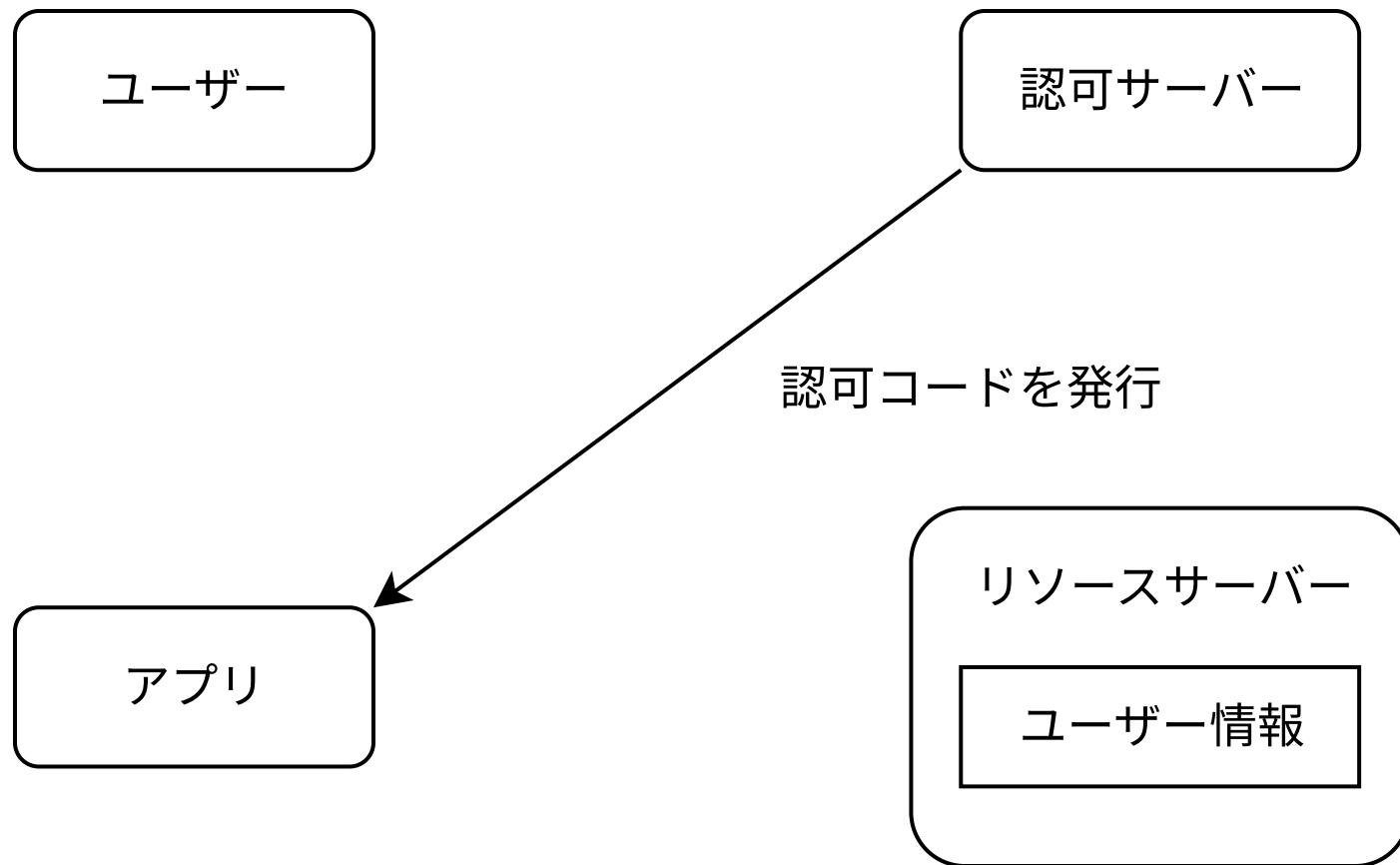
認証、認可フロー



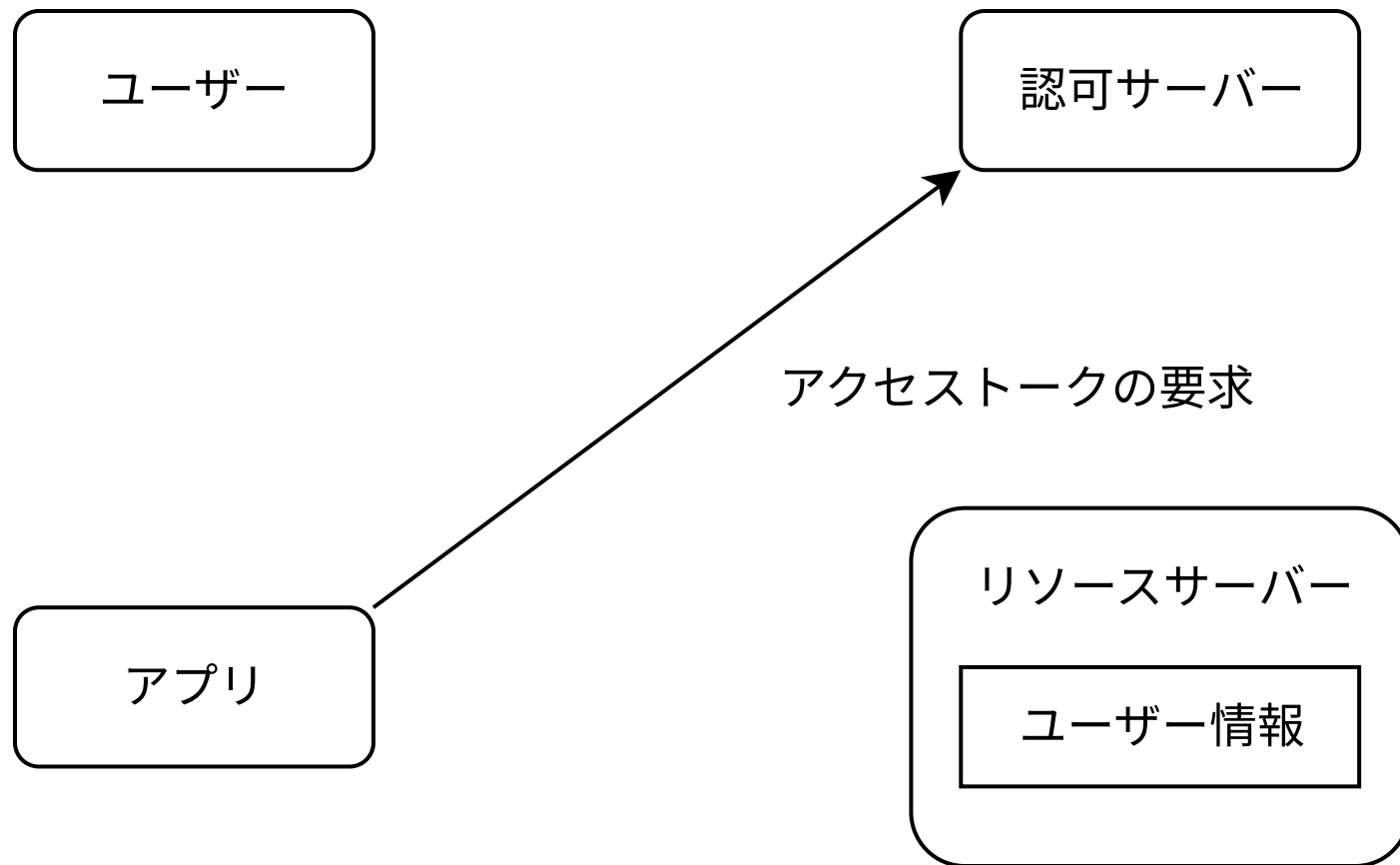
認証、認可フロー



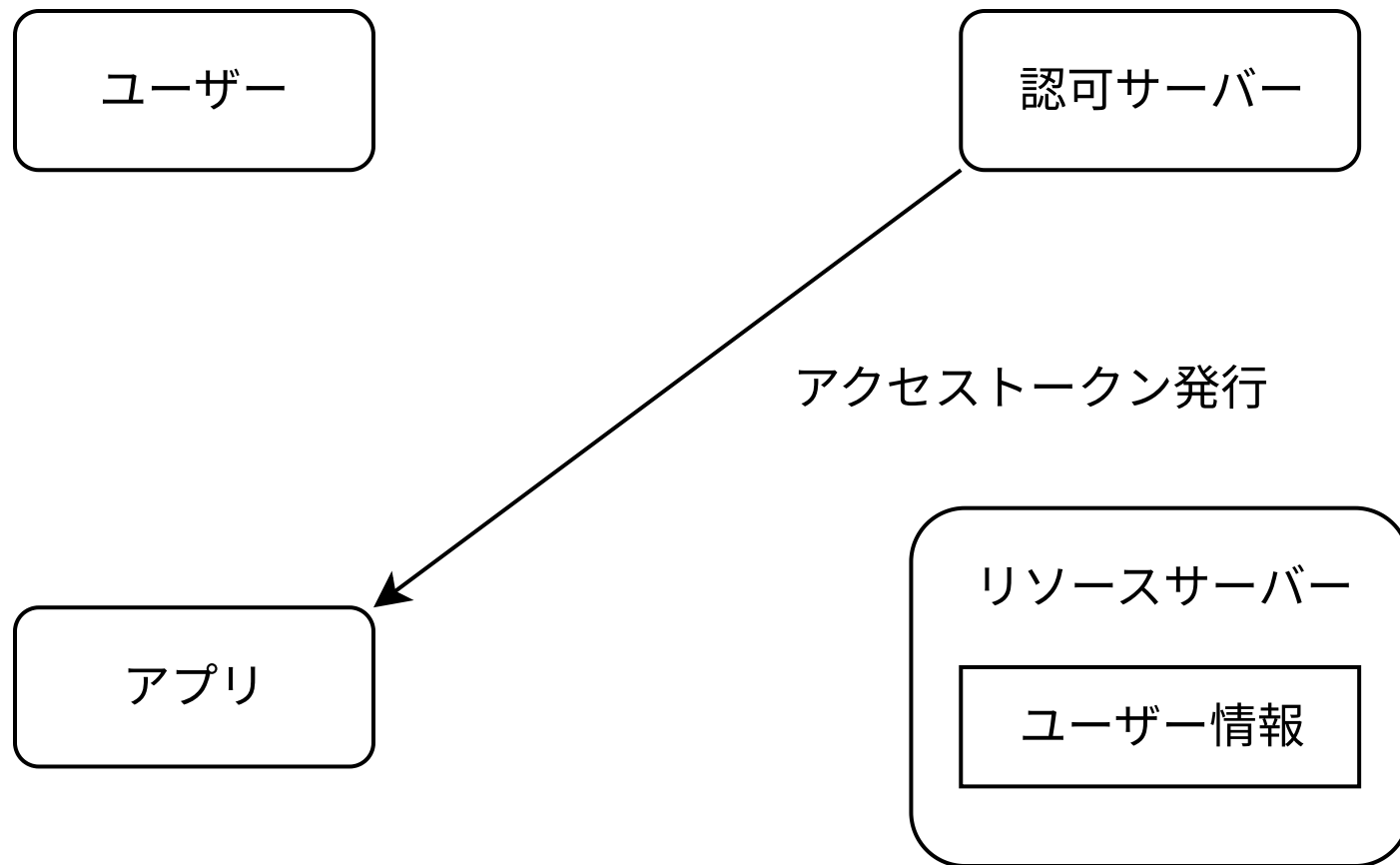
認証、認可フロー



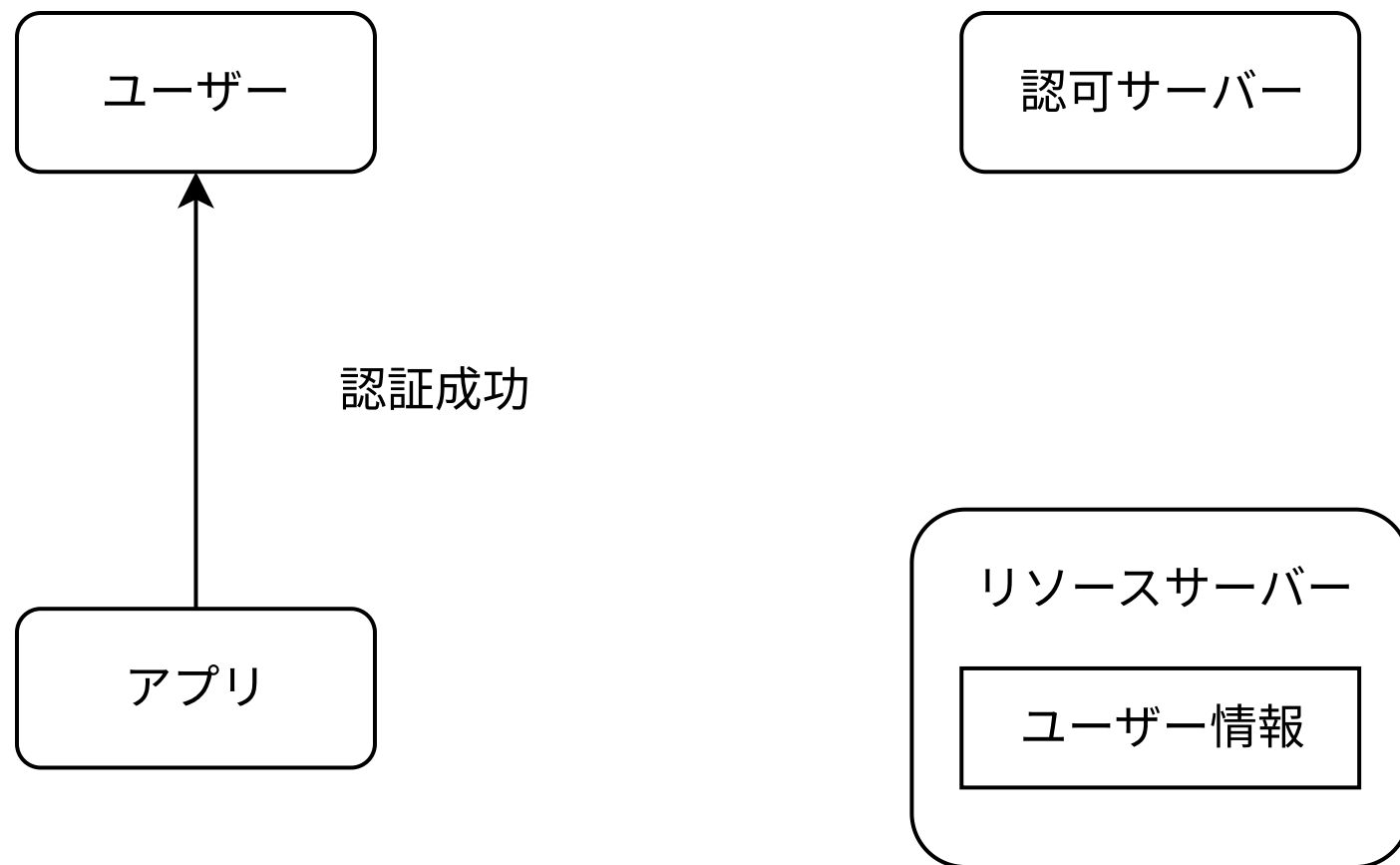
認証、認可フロー



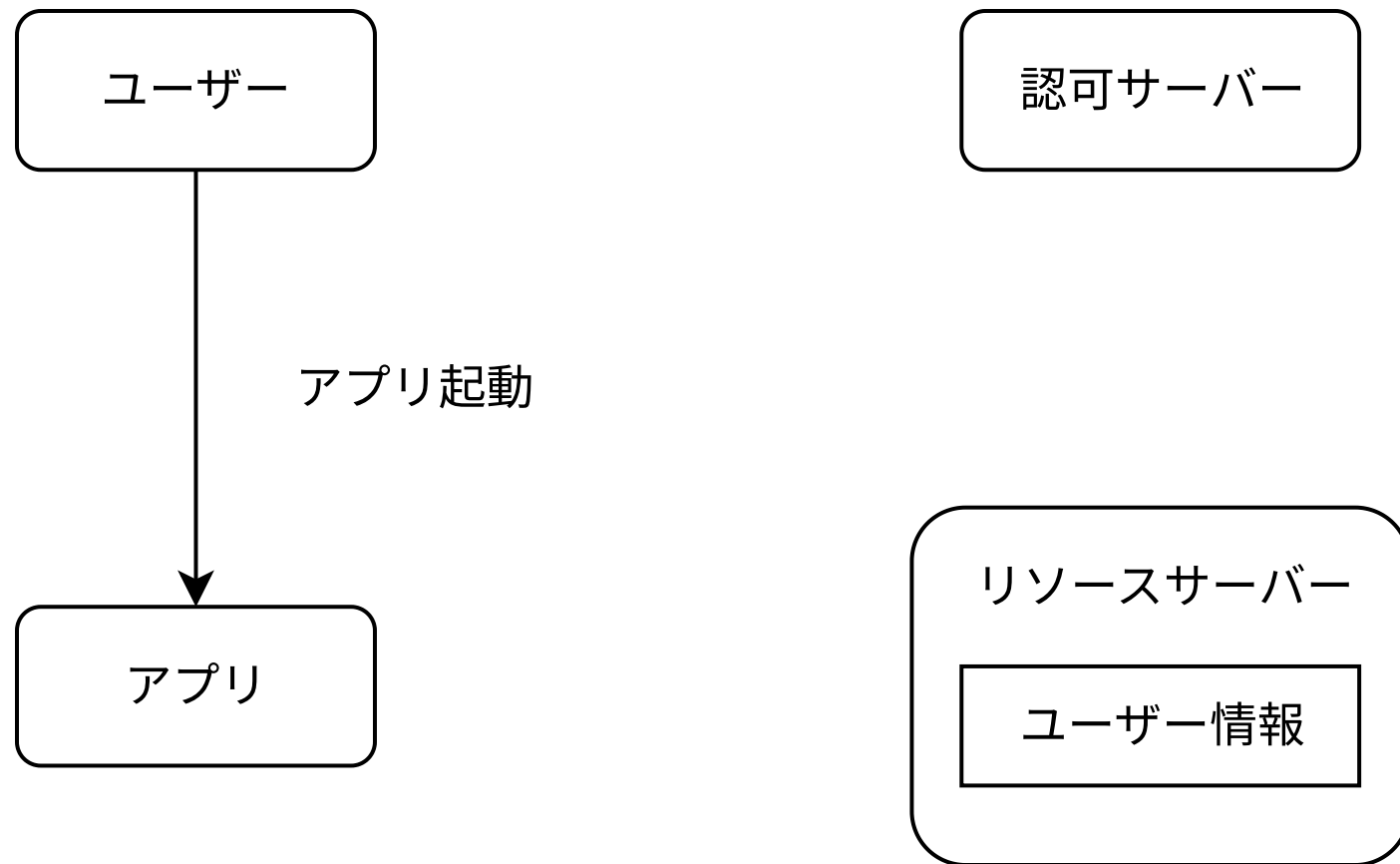
認証、認可フロー



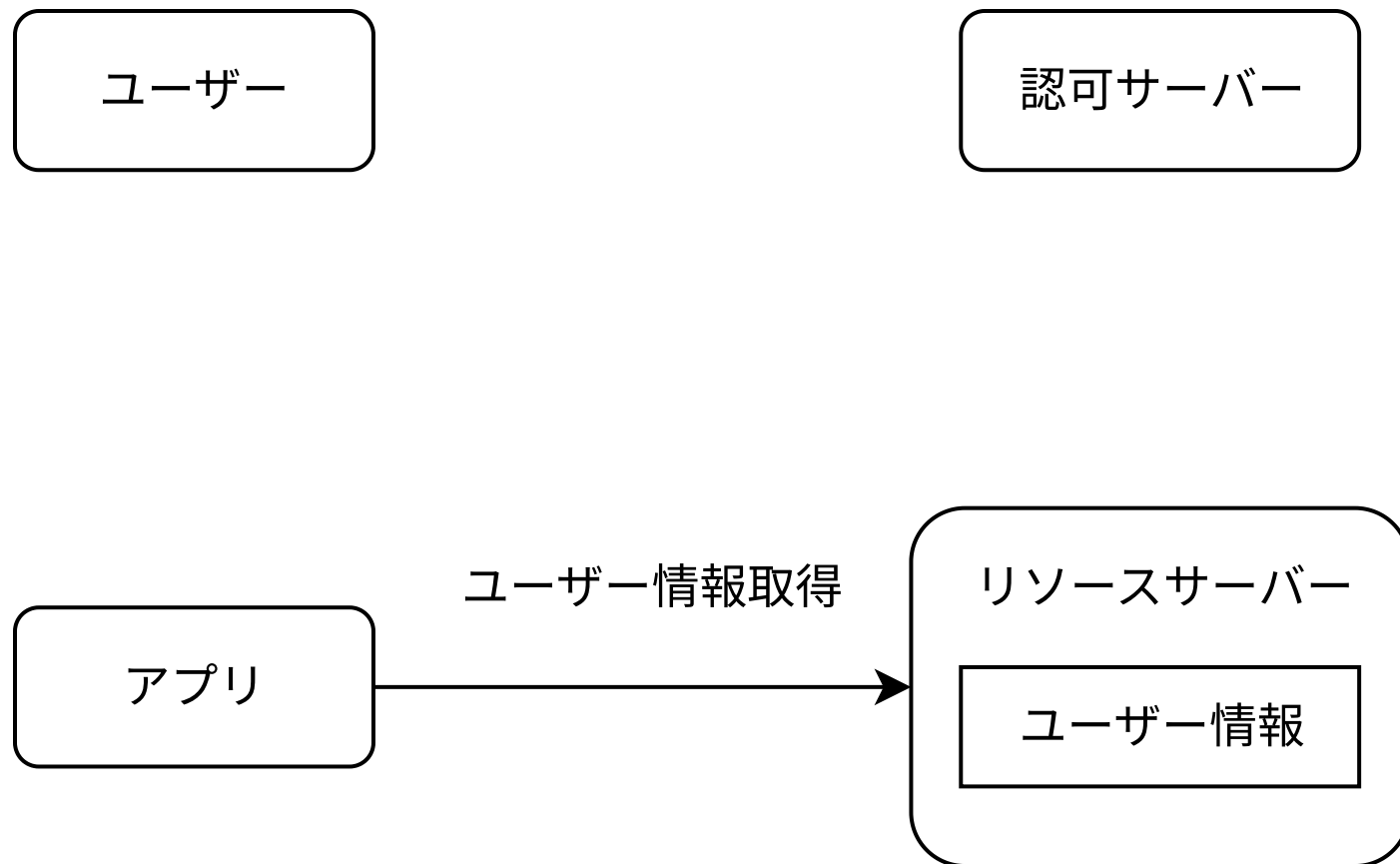
認証、認可フロー



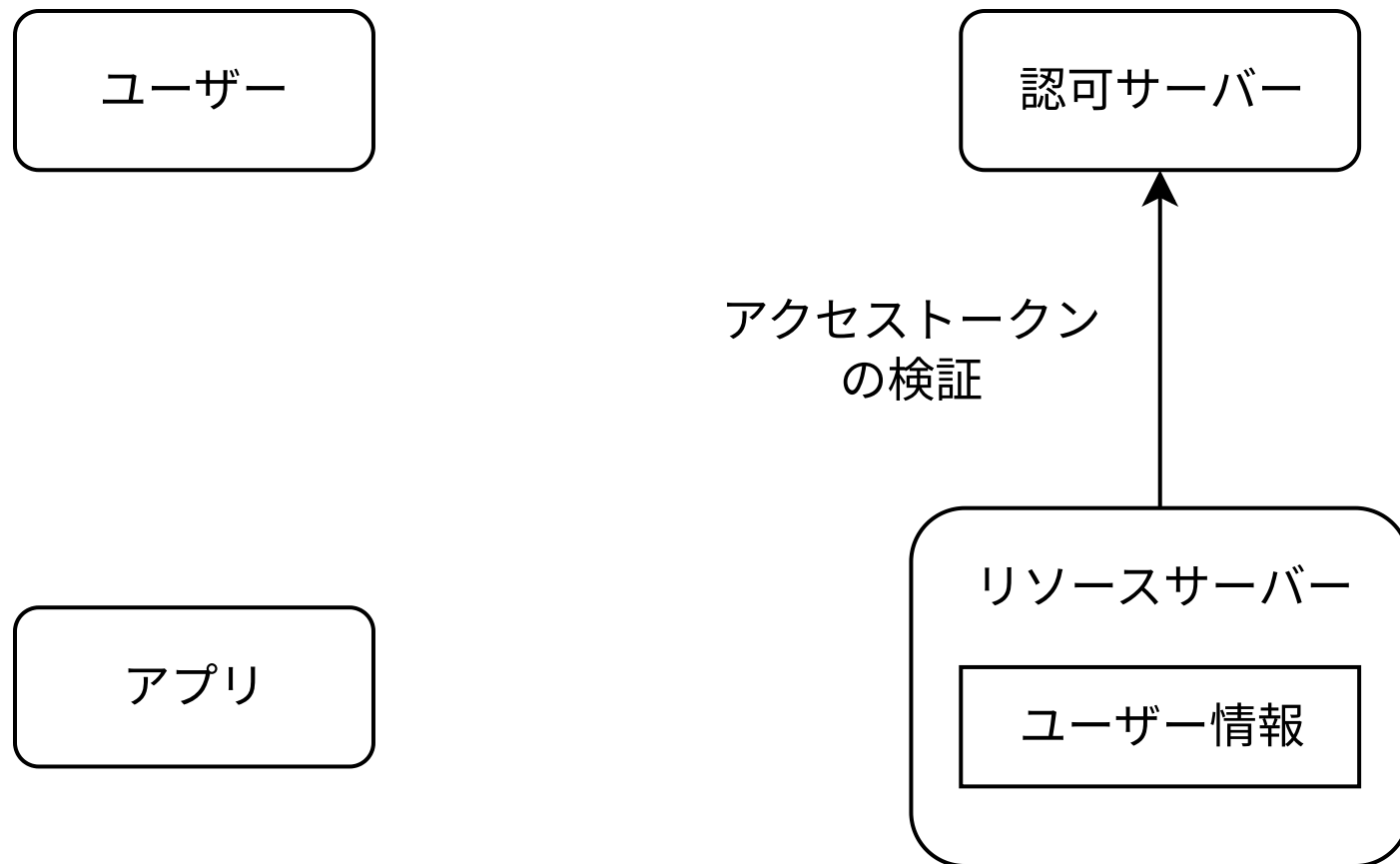
リソース取得フロー



リソース取得フロー



リソース取得フロー



リソース取得フロー

ユーザー

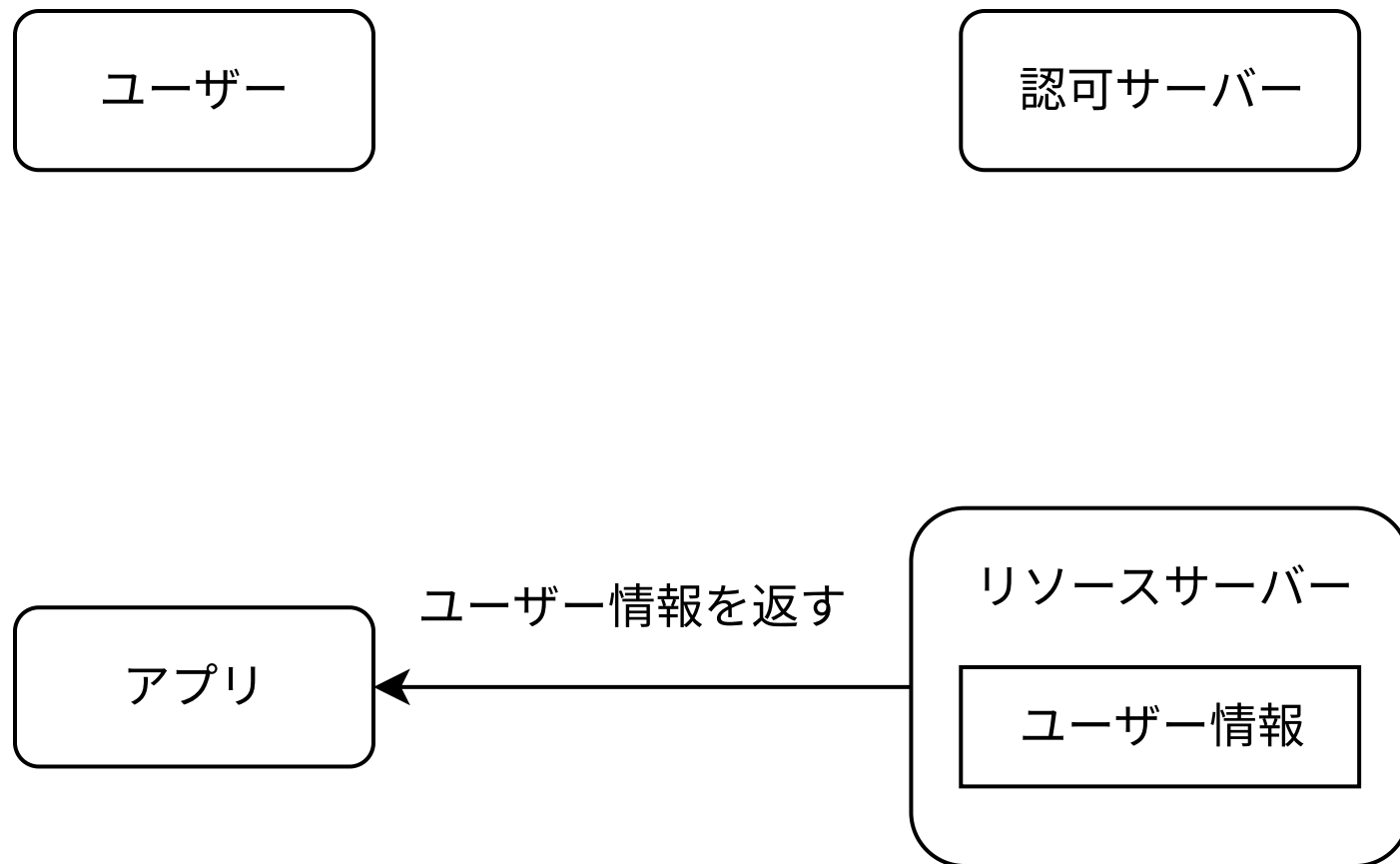
アプリ

認可サーバー

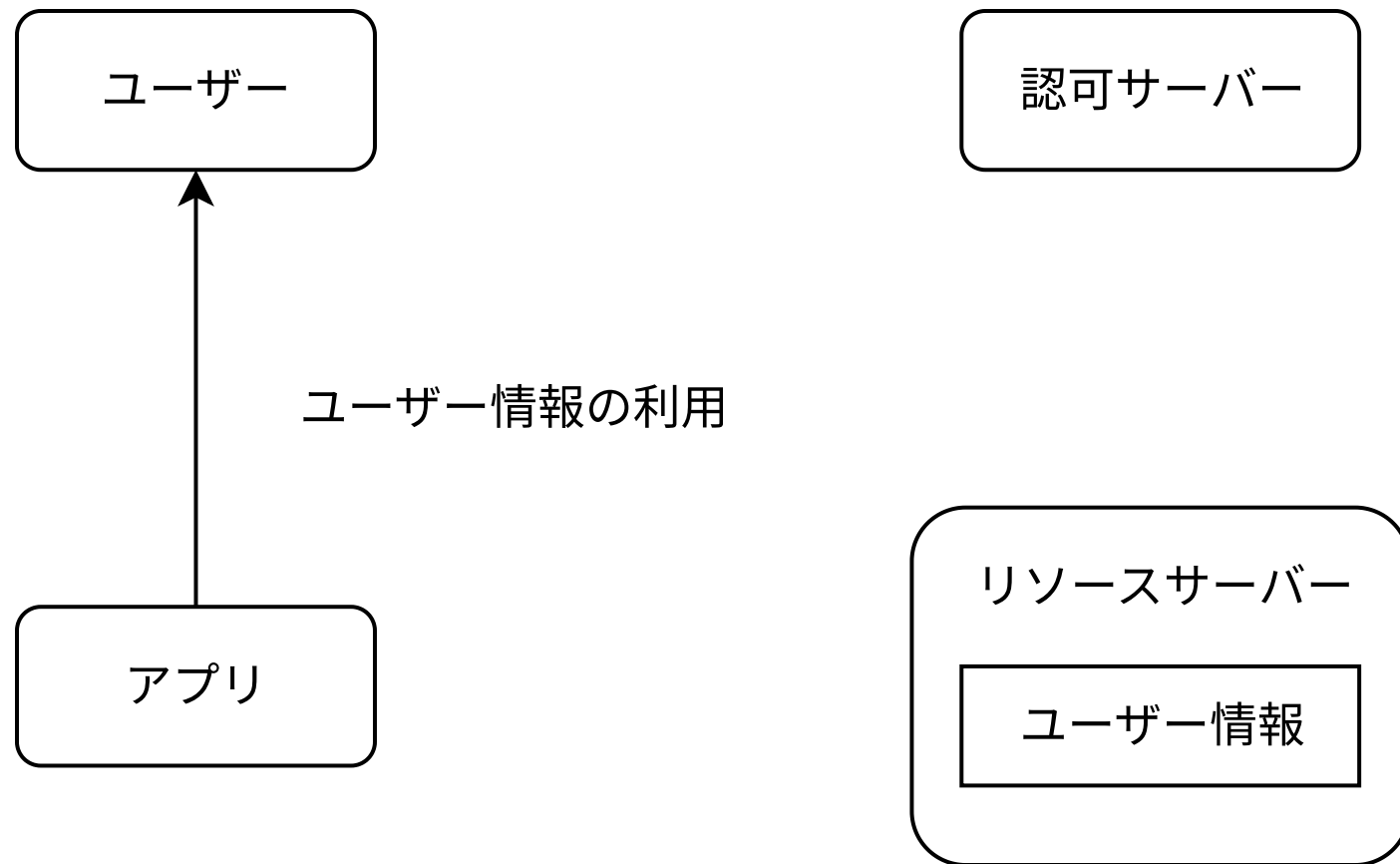
検証成功 (email)

リソースサーバー
ユーザー情報

リソース取得フロー



リソース取得フロー



使用ツール

- Ruby on Rails：リソースサーバー
- Ory Hydra：認可サーバー
- Node.js：クライアントアプリケーション

デモ

クライアントアプリケーション (Node.js)

- index.js

認証、認可フロー画面 (Ruby on Rails)

- oauth_login_controller.rb (ログイン画面ロジック)
- oauth_login_show.html.erb (ログイン画面ビュー)
- oauth_consent_controller.rb (認可画面ロジック)
- oauth_consent_show.html.erb (認可画面ビュー)

ユーザー情報取得API (Ruby on Rails)

- api_user_controller.rb

Hydraリクエスト (Ruby on Rails)

- hydra_service.rb