

간편인증API 사용자 매뉴얼

2022년 01월 17일

* 개정 이력

순번	요건번호	개(제)정일자	Rev	개(제)정 내용요약	비고
1		2021.10.19	v1.0	신규생성	
2		2021.11.01	v2.0	사용 예제 추가	
3		2021.12.03	v2.1	중간 로직 및 토큰 유효 시간	
4		2021.12.24	v3.0	국민건강보험공단 및 정부24 통합	
5		2022.01.17	v3.1	간편인증 요청 API 호출 URL 추가	

목차

1. 간편 인증이란?	4
2. 간편 인증 절차 및 사용법	4
2.1. 간편 인증 요청 API 호출	4
2.2. 모바일에서 인증 처리	4
2.3. 각종 서비스 API 호출	5
2.4. 중간 로직	5
2.5. 토큰 유효 시간	5
2.6. 샘플 코드	5
2.7. 사용 예제 – 국민건강보험공단 간편인증 요청(C#)	6
2.8. 사용 예제 – 국민건강보험공단 건강검진내역(C#)	7
2.9. 사용 예제 – 정부 24 간편인증 요청(NodeJS)	8
2.10. 사용 예제 – 정부 24 주민등록증 진위여부 확인(NodeJS)	9

1. 간편 인증이란?

인증서를 사용하지 않고 카카오톡, 페이코, 국민은행모바일, 삼성Pass, 텔레콤Pass 등을 통하여 모바일로 간편하게 인증하여 로그인하는 방식입니다.

2. 간편 인증 절차 및 사용법



2.1. 간편 인증 요청 API 호출

API 를 통해 간편 인증 요청

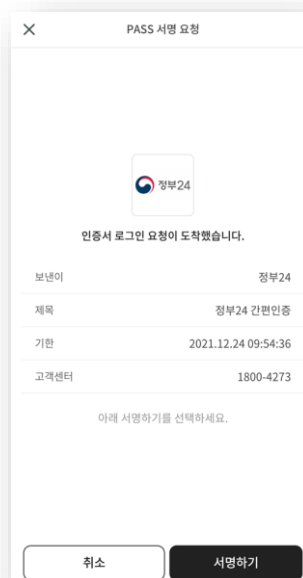
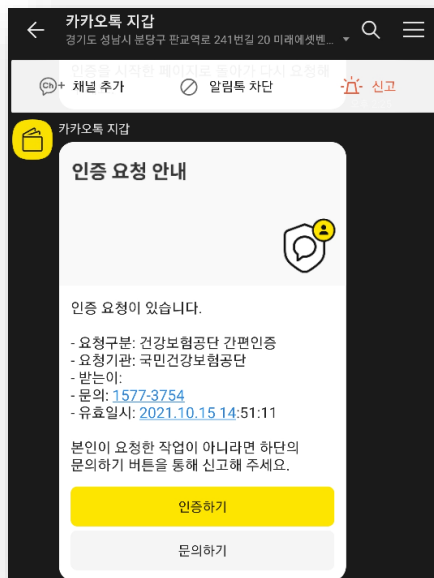
- 국민건강보험공단

<https://tilko.net/Help/Api/POST-api-apiVersion-NhisSimpleAuth-SimpleAuthRequest>

- 정부24

<https://tilko.net/Help/Api/POST-api-apiVersion-GovSimpleAuth-SimpleAuthRequest>

2.2. 모바일에서 인증 처리



2.3. 각종 서비스 API 호출

- 국민건강보험공단 건강검진내역

<https://tilko.net/Help/Api/POST-api-apiVersion-NhisSimpleAuth-Ggpab003M0105>

- 정부24 주민등록증 진위여부 확인

<https://tilko.net/Help/Api/POST-api-apiVersion-GovSimpleAuth-AA090UserJuminCheckResApp>

이때, 반드시 모바일에서 인증 완료 후 사용하셔야 하며, 같은 서비스에도 인증서용 API와 간편인증용 API 등으로 구분되어 있으니 유의하시기 바랍니다.

2.4. 중간 로직



API 와 API 사이에는 모바일에서의 인증 처리 과정이 있습니다. 이 과정은 사람이 직접 인증해야 하는 단계이므로, 서버에서는 사람이 언제 인증을 완료했는지 알 수 없습니다. 따라서, 인증을 완료했다는 것을 서버에 알려주기 위한 장치를 별도로 마련하셔야 합니다.

예를 들어, 인증 완료 버튼을 만들고 해당 버튼을 누르면 실제 사용할 서비스 API를 호출하도록 할 수 있습니다.

2.5. 토큰 유효 시간

카카오톡, 페이코 등의 간편인증 서비스의 정책에 따라 제한 시간이 다를 수 있습니다. 일정 시간 내에 인증하지 않으면 세션이 종료됩니다. 그 이후에 실제 서비스 API를 호출하시면 '토큰 유효 시간 만료' 메시지가 반환되며, 원하는 정보를 얻을 수 없게 됩니다.

2.6. 샘플 코드

GitHub 사이트를 통해 프로그래밍 언어별 샘플 코드를 제공해드립니다.

<https://github.com/Tilkoblet?tab=repositories>

2.7. 사용 예제 – 국민건강보험공단 간편인증 요청(C#)

```
using Microsoft.VisualStudio.TestTools.UnitTesting;
using System;
using System.Diagnostics;
using System.IO;

namespace UnitTest.KR.OR.NHIS
{
    [TestClass]
    public class 간편인증요청
    {
        [TestMethod]
        public void TILKO_API()
        {
            // API 상세설명 URL
            // 간편인증: https://tilko.net/Help/Api/POST-api-apiVersion-NhisSimpleAuth-SimpleAuthRequest

            try
            {
                Tilko.API.REST _rest = new Tilko.API.REST(Constant.ApiKey);
                _rest.Init();

                // 국민건강보험공단의 간편인증요청 endPoint 설정
                _rest.SetEndPointUrl(Constant.ApiHost + "/api/v1.0/nhissimpleauth/simpleauthrequest");

                // Body 추가
                _rest.AddBody("PrivateAuthType", "0", false);
                _rest.AddBody("UserName", "홍길동", true);
                _rest.AddBody("BirthDate", "20210101", true);
                _rest.AddBody("UserCellphoneNumber", "01012345678", true);

                // API 호출
                string _result = _rest.Call();
                if (_rest.HttpStatusCode != System.Net.HttpStatusCode.OK)
                {
                    throw new Exception(_rest.Message);
                }
            }
            catch (Exception err)
            {
                Debug.WriteLine(err.Message);
            }
        }
    }
}
```

2.8. 사용 예제 – 국민건강보험공단 건강검진내역(C#)

```
using Microsoft.VisualStudio.TestTools.UnitTesting;
using System;
using System.Diagnostics;
using System.IO;

namespace UnitTest.KR.OR.NHIS
{
    [TestClass]
    public class 건강검진내역
    {
        [TestMethod]
        public void TILKO_API()
        {
            // API 상세설명 URL
            // 건강검진내역: https://tilko.net/Help/Api/POST-api-apiVersion-Nhis-Ggpab003M0105

            try
            {
                // 간편인증 응답 및 로그인 처리(모바일에서 인증 완료 후, 진행하셔야 합니다.)

                TILKO.API.REST _rest = new TILKO.API.REST(Constant.ApiKey);
                _rest.Init();

                // 국민건강보험공단의 건강검진내역 endPoint 설정
                _rest.SetEndPointUrl(Constant.ApiHost + "/api/v1.0/nhis/ggpab003m0105");

                // Body 추가
                _rest.AddBody("CxId", "{간편인증 응답의 CxId 값}", false);
                _rest.AddBody("PrivateAuthType", "{간편인증 응답의 PrivateAuthType 값}", false);
                _rest.AddBody("ReqTxId", "{간편인증 응답의 ReqTxId 값}", false);
                _rest.AddBody("Token", "{간편인증 응답의 Token 값}", false);
                _rest.AddBody("TxId", "{간편인증 응답의 TxId 값}", false);
                _rest.AddBody("UserName", "{간편인증 응답의 UserName 값}", false);
                _rest.AddBody("BirthDate", "{간편인증 응답의 BirthDate 값}", false);
                _rest.AddBody("UserCellphoneNumber", "{간편인증 응답의 UserCellphoneNumber 값}", false);
                _rest.AddBody("NhisQuery", "0", false); // 검색 전체: 0 / 직장가입자: 1 / 지역가입자: 2 / 가입자 전체: 3

                // API 호출
                string _result = _rest.Call();
                if (_rest.HttpStatusCode != System.Net.HttpStatusCode.OK)
                {
                    throw new Exception(_rest.Message);
                }
            }
            catch (Exception err)
            {
                Debug.WriteLine(err.Message);
            }
        }
    }
}
```

2.9. 사용 예제 – 정부24 간편인증 요청(NodeJS)

```
const fs = require("fs");
const Rest = require("../../Tilko.API/RESTLocal").Tilko.API.REST;
const Constant = require("../../UnitTest/ConstantLocal").UnitTest.Constant;

var Tilko = Tilko || {};

(function (_Tilko) {

    // API 상세설명 URL
    // https://tilko.net/Help/Api/POST-api-apiVersion-GovSimpleAuth-SimpleAuthRequest

    try {
        let _rest = new Rest(Constant.ApiKey);
        _rest.Init();

        // 간편인증 요청 endPoint 설정
        _rest.SetEndPointUrl(Constant.ApiHost + "api/v1.0/GovSimpleAuth/SimpleAuthRequest");

        // Body 추가
        _rest.AddBody("PrivateAuthType", "4"); //인증종류 0:카카오톡 1:페이코 2:국민은행모바일 3:삼성패스 4:텔레콤Pass
        _rest.AddBody("UserName", "홍길동", true); // [암호화] 이용자명
        _rest.AddBody("BirthDate", "20210101", true); // [암호화] 생년월일 (yyyyMMdd)
        _rest.AddBody("UserCellphoneNumber", "01012345678", true); // [암호화] 휴대폰번호

        // API 호출
        const Response = _rest.Call();
        console.log("Response:", Response);
    } catch (e) {
        console.error(e);
    }

})(Tilko);
```

2.10. 사용 예제 – 정부24 주민등록증 진위여부 확인(NodeJS)

```

const FS = require("fs");
const Rest = require(".././../Tilko.API/RESTLocal").Tilko.API.REST;
const Constant = require(".././../UnitTest/ConstantLocal").UnitTest.Constant;

var Tilko = Tilko || {};

(function (Tilko) {

    // API 상세설명 URL
    // https://tilko.net/Help/Api/POST-api-apiVersion-GovSimpleAuth-AA090UserJuminCheckResApp

    try {
        // 간편인증 응답 및 로그인 처리(모바일에서 인증 완료 후, 진행하여야 합니다.)
        let _rest = new Rest(Constant.ApiKey);
        _rest.Init();

        // 건강보험공단의 건강검진내역 조회 endPoint 설정
        _rest.SetEndPointUrl(Constant.ApiHost + "api/v1.0/GovSimpleAuth/AA090UserJuminCheckResApp");

        // 간편인증 요청 후 받은 값들
        const RequestLogin = {
            "CxId": "",
            "PrivateAuthType": "",
            "ReqTxId": "",
            "Token": "",
            "TxId": "",
            "UserName": "",
            "BirthDate": "",
            "UserCellphoneNumber": "",
        };

        // Body 추가
        _rest.AddBody("CxId", RequestLogin.CxId, false);
        _rest.AddBody("PrivateAuthType", RequestLogin.PrivateAuthType, false);
        _rest.AddBody("ReqTxId", RequestLogin.ReqTxId, false);
        _rest.AddBody("Token", RequestLogin.Token, false);
        _rest.AddBody("TxId", RequestLogin.TxId, false);
        _rest.AddBody("UserName", RequestLogin.UserName, true);
        _rest.AddBody("BirthDate", RequestLogin.BirthDate, true);
        _rest.AddBody("UserCellphoneNumber", RequestLogin.UserCellphoneNumber, true);
        _rest.AddBody("PersonName", "김길동", true);
        _rest.AddBody("IdentityNumber", "9012311234567", true);
        _rest.AddBody("PublishDate", "20200101", false);

        // API 호출
        const Response = _rest.Call();
        console.log("Response:", Response);
    } catch (e) {
        console.error(e);
    }

})(Tilko);

```

//간편인증 요청 후 받은 CxId
 //간편인증 요청 후 받은 PrivateAuthType
 //간편인증 요청 후 받은 ReqTxId
 //간편인증 요청 후 받은 Token
 //간편인증 요청 후 받은 TxId
 //[암호화] 간편인증 요청 후 받은 UserName
 //[암호화] 간편인증 요청 후 받은 BirthDate
 //[암호화] 간편인증 요청 후 받은 UserCellphoneNumber
 //[암호화] 조회 대상자의 성명
 //[암호화] 조회 대상자의 주민등록번호(9001011234567)
 //주민등록증 발행일(yyyyMMdd)