

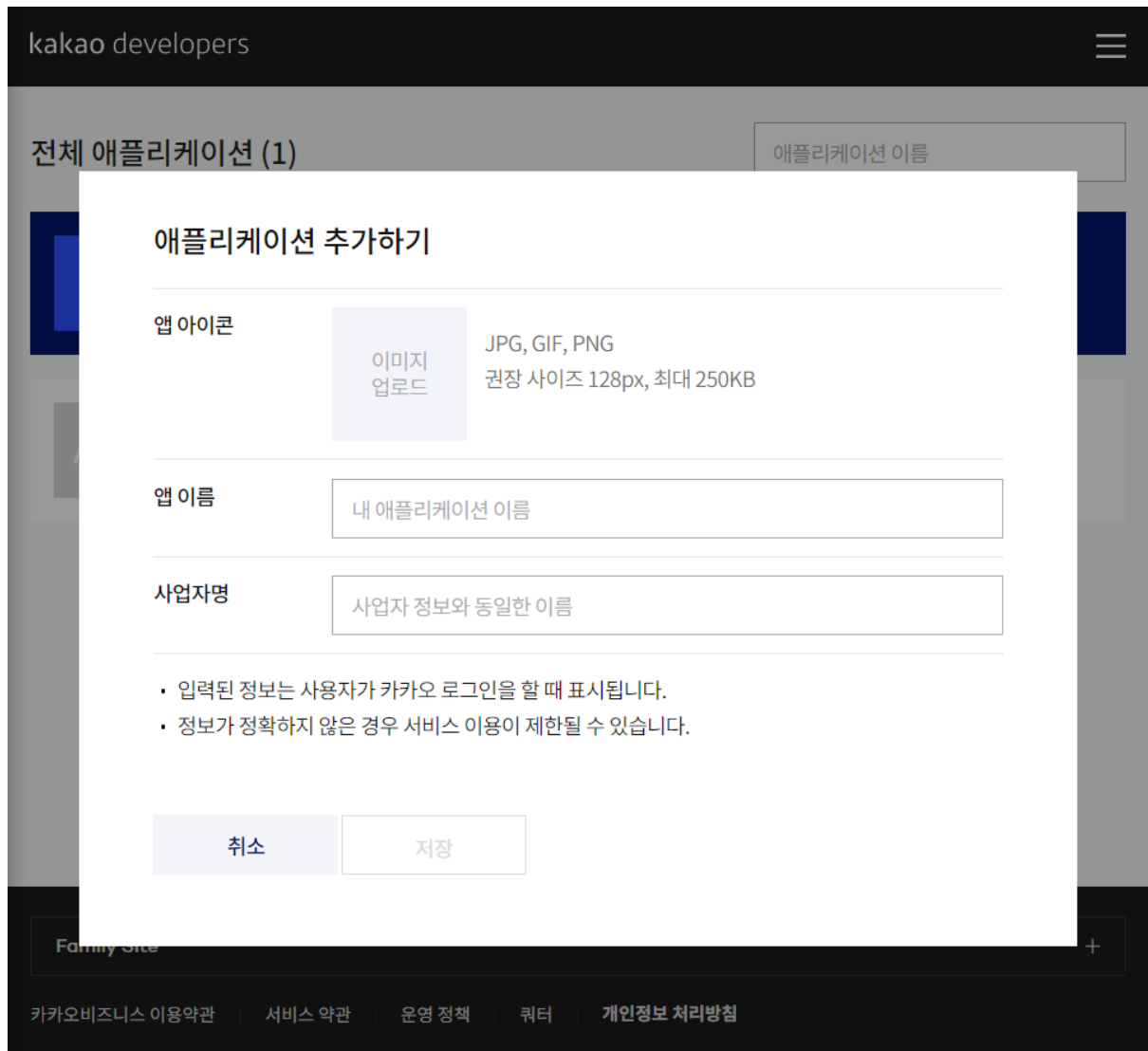
# 외부 서비스 사용 문서

## Kakao OAuth2 사용하여 소셜 로그인 구현

### 애플리케이션 등록

KaKao 소셜 로그인을 사용하기 위해서는

<https://developers.kakao.com/> 에 접속하여 내 애플리케이션을 추가합니다.



The screenshot shows the 'kakao developers' website with a modal dialog titled '애플리케이션 추가하기' (Add Application). The dialog contains the following fields and information:

- 앱 아이콘** (App Icon): A placeholder box with the text '이미지 업로드' (Upload Image). To the right, it specifies 'JPG, GIF, PNG' and '권장 사이즈 128px, 최대 250KB' (Recommended size 128px, maximum 250KB).
- 앱 이름** (App Name): A text input field with the placeholder '내 애플리케이션 이름' (My Application Name).
- 사업자명** (Business Name): A text input field with the placeholder '사업자 정보와 동일한 이름' (Name same as business information).

Below the input fields, there are two bullet points:

- 입력된 정보는 사용자가 카카오 로그인을 할 때 표시됩니다.
- 정보가 정확하지 않은 경우 서비스 이용이 제한될 수 있습니다.

At the bottom of the dialog, there are two buttons: '취소' (Cancel) and '저장' (Save).

The background of the website shows a header with 'kakao developers' and a hamburger menu. Below the header, there is a section titled '전체 애플리케이션 (1)' (All Applications (1)) with a search bar labeled '애플리케이션 이름' (Application Name). At the bottom of the page, there is a footer with links: '카카오비즈니스 이용약관' (Kakao Business Terms of Service), '서비스 약관' (Service Terms of Service), '운영 정책' (Operating Policy), '쿼터' (Quota), and '개인정보 처리방침' (Privacy Policy).

### 플랫폼 등록

내 애플리케이션>플랫폼에서 사이트 도메인을 등록합니다.

도메인으로 등록되지 않은 주소에서 요청을 받으면 기능 실행이 되지 않기 때문에

테스트와 배포를 진행하는 모든 도메인을 등록해줍니다.

### Web 플랫폼 수정

---

#### 사이트 도메인

JavaScript SDK, 카카오톡 공유, 카카오톡, 메시지 API 사용시 등록이 필요합니다.  
여러개의 도메인은 줄바꿈으로 추가해주세요. 최대 10까지 등록 가능합니다. 추가 등록은 포럼(데브톡)으로 문의주세요.  
예시: (O) `https://example.com` (X) `https://www.example.com`

`http://localhost:8080`  
`http://localhost:8081`  
`http://localhost:8082`  
`https://j7a604.p.ssafy.io`

#### 기본 도메인

기본 도메인은 첫 번째 사이트 도메인으로, 카카오톡 공유와 카카오톡 메시지 API를 통해 발송되는 메시지의 Web 링크 기본값으로 사용됩니다.

`http://localhost:8080`

취소

저장

## Redirect URI 등록

내 애플리케이션>플랫폼>redirect uri 등록하러 가기를 눌러

카카오 로그인 페이지에서 redirect할 나의 서비스의 도메인 주소를 입력합니다.

독으로 서비스를 시작할 수 있습니다.  
니다.

### Redirect URI

#### Redirect URI

카카오 로그인에서 사용할 OAuth Redirect URI를 설정합니다.

여러개의 URI를 줄바꿈으로 추가해주세요. (최대 10개)

REST API로 개발하는 경우 필수로 설정해야 합니다.

예시: (O) <https://example.com/oauth> (X) <https://www.example.com/oauth>

```
http://localhost:8081  
http://localhost:8081/login  
http://localhost:8082  
http://j7a604.p.ssafy.io  
https://j7a604.p.ssafy.io  
https://j7a604.p.ssafy.io/login
```

취소

저장

스 토큰과 함께 발급받을

kaoLogin

## JavaScript SDK 파일 포함 및 초기화

- Vue 프로젝트의 index.htm l파일에 아래 url의 scripts를 삽입합니다.

<https://developers.kakao.com/docs/latest/ko/getting-started/sdk-js>

- Vue 프로젝트의 main.js 파일에 아래 코드를 삽입합니다.

(이때 'JAVASCRIPT\_KEY' 안에 자신이 발급받은 애플리케이션의 JavaScript 키를 넣어줍니다.)

```
Kakao.init('JAVASCRIPT_KEY');  
Kakao.isInitialized();
```

## 로그인 요청 및 토큰 받기

카카오 로그인 버튼을 눌렀을 때 간편 로그인을 요청하는 함수입니다.

원하는 위치에 삽입하여 간편 로그인을 요청합니다.

```
Kakao.Auth.authorize()
```

# kakao

카카오메일 아이디, 이메일, 전화번호

비밀번호

☒ 로그인 상태 유지

로그인

또는

QR코드 로그인

회원가입      계정 찾기 | 비밀번호 찾기

[이용약관](#) [개인정보 처리방침](#) [운영정책](#) [고객센터](#) [공지사항](#) [한국어](#) ^

Copyright © Kakao Corp. All rights reserved.

## Cookie 생성

Vue 프로젝트에서 vue-cookies를 설치합니다.

```
npm install vue-cookies
```

main.js에 import 문을 삽입하여 cookie를 생성, 삭제, 값 가져오기 등을 사용할 수 있습니다.

```
import vueCookies from "vue-cookies";
```

## 구현 Code

발급 받은 앱키와 설정한 redirect uri를 사용하여 카카오 소셜 로그인 기능을 구현합니다.

저희 팀의 경우, 설정한 환경으로 인해

FrontEnd에서 로그인을 확인하고, Cookie에 값을 전달해 그 값을

RequestHeader에 넣어 Backend와 주고 받습니다.

```
const KakaoAPI = {
  async KakaoLogin() {
    let code = new URL(window.location.href).searchParams.get("code");
    if (code) {
      axios
        .post(
          "https://kauth.kakao.com/oauth/token?grant_type=authorization_code&client_id"
          +
          ClientId +
          "&redirect_uri=" +
          KakaoRedirectURL +
          "&code=" +
          code +
          "&client_secret=" +
          ClientSecret,
          {
            headers: {
              "Content-Type": "application/x-www-form-urlencoded",
            },
          },
        )
        .then((res) => {
          axios
            .get("https://kapi.kakao.com/v2/user/me", {
              headers: {
                "Content-Type":
                  "application/x-www-form-urlencoded;charset=utf-8",
                Authorization: "Bearer " + res.data.access_token,
              },
            })
            .then((res) => {
              api
```

```

        .Login(res.data.id, res.data.kakao_account.profile.nickname)
        .then((res) => {
            document.cookie = "userId=" + res.userId;
            document.cookie = "nickname=" + res.nickname;
            window.location.replace(MolaengURL);
        });
    });
});
},
KakaoAuth() {
    window.Kakao.Auth.authorize({
        redirectUri: KakaoRedirectURL,
    });
},
};
};

```

## 실행 결과

