

REPORT



부산대학교
PUSAN NATIONAL UNIVERSITY

텀프로젝트 최종보고서 : 사진 변환 및 저장 서비스

1조

201661701 강동민

201524495 안준수

201424534 정지훈

목차

- 1. 개발개요
- 2. 개발결과 - Android UI
- 3. 개발결과 - Imgur API 기능
- 4. 개발결과 - 이미지 변환 API
- 5. 개발결과 - Google AR Core 기능

1. 개발개요

1.1 사진 공유 서비스에서의 Cloud 기능의 수요

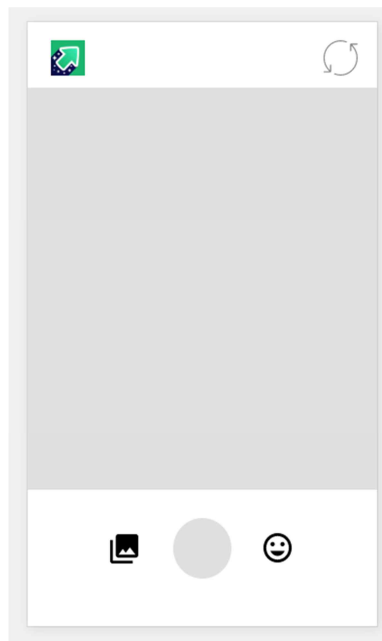
요즘 모바일 어플리케이션 시장에서는 인스타그램이나 페이스북 등 사진을 공유하는 서비스가 인기가 높다. 그런데, 점점 스마트폰 카메라의 성능이 커지면서 사진의 용량 또한 증가하게 되었다. 이는 대용량 사진의 계속된 저장으로 스마트폰의 로컬 저장소를 부족하게 만들어 디바이스의 성능을 저하시킬 수 있다. 따라서, 사진의 공유를 위해서 Cloud 기능을 도입하여 제공하는 것이 Application이 효율적 디바이스 관리를 도모할 수 있는 Application이 될 것이다.

특히나, 본 프로젝트가 제공하는 기능과 관련된, 일회적이고 유희적인, 재미난 사진의 경우에는 로컬에 오래 저장하기 보다, 장기적으로 필요 없는 것은 Cloud를 통해 잠시 저장을 하도록 하고, 마음에 드는 것만 로컬에 저장하는 기능을 도입한다면 효율적일 것이다.

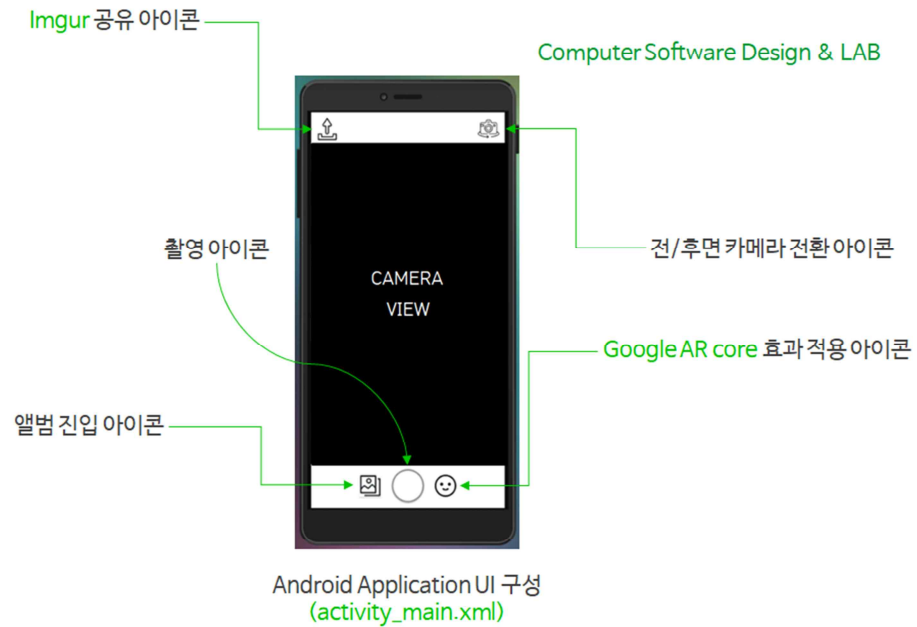
1.2 사진 공유 서비스에서의 익명성의 수요

한편, 인스타그램과 같은 SNS와는 반면에, Telegram과 같이 익명성이 보장되는 어플리케이션에 대한 수요 또한 상당하다. 고용량 사진 클라우드 저장에 대한 수요와 더불어 익명성에 대한 수요 또한 충족시킬 수 있어야 한다는 것이다. 일회적이거나 유희적인, 재미난 meme과 같은 일명 ‘짤’ 과 같은 사진들은 Cloud에 저장할 때에 익명성 또한 중요할 것이다. 따라서 사진 공유 서비스를 제공하되, Cloud 기능과 익명적 기능 모두를 만족하는 어플리케이션을 구성하려 하였다.

2. 개발결과 - Android UI



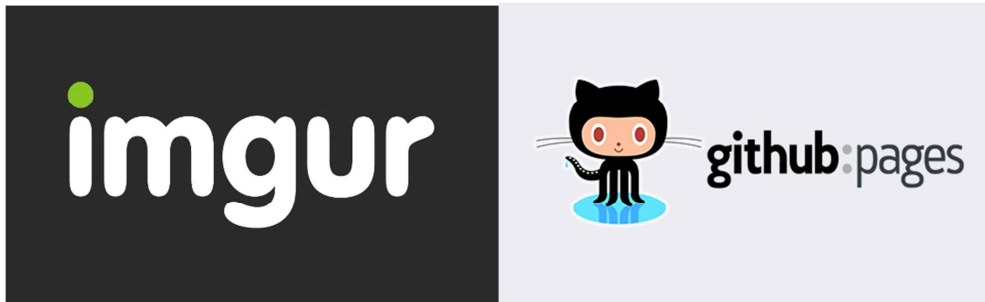
(기존 목표한 User Interface)



(실제 구현한 User Interface)

Android UI의 경우 XML을 통하여 제작하였다. 이미 시장에 출시된 사진 변환 서비스 어플리케이션 (Snapchat, SNOW, B612, FaceApp 등)을 참고하였다. 사진변환 서비스 어플리케이션 중 대다수가 위와 같은 UI를 채택하고 있으며, 효율적이고 직관적인 UI라 할 수 있다.

3. 개발결과 - Imgur API 기능



Imgur는 이미지 저장 클라우드 호스팅 서비스이다. 여러 다양한 사진을 Imgur 서버에 업로드하여 공유하거나 저장할 수 있다. Imgur는 API도 제공하는데, 이를 Application에 적용하였다.

특히, Imgur는 본 어플리케이션의 개발 목적인 Cloud 제공 기능 및 익명성을 보장할 수 있다.

Imgur API의 적용은 HTML과 JavaScript를 이용하였다. HTML/JavaScript로 구성한 웹페이지를 Github에서 제공하는 일종의 간편한 웹호스팅 서비스인 Github pages를 이용하였다. 이를 통해 복잡한 서버 구축없이 편리한 개발이 가능하였다.

이후 안드로이드 Application 상에서, webview를 통하여 웹페이지를 제공하는 식으로 구성하였다.



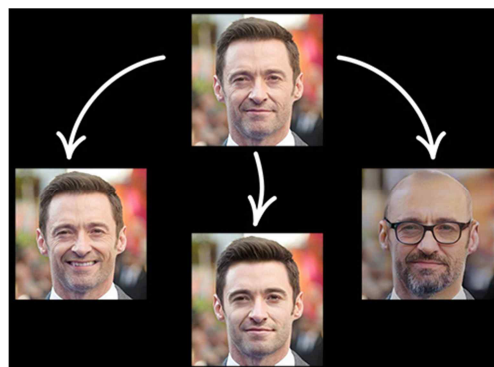
원하는 이미지를 선택하면 Imgur 서버에 자동으로 사진이 업로드 되며, 성공적인 업로드 시에는 다음과 같이 알람창이 뜨며 이후 문자(SMS), 트위터, 페이스북, 인스타그램 등 다양한 어플리케이션으로 공유할 수 있으며, 공유링크 또한 간편하게 클립보드에 복사할 수 있다.



또, 데스크톱과 달리 마우스 커서가 없는 모바일을 위해 간단한 CSS를 적용하여, 버튼 터치 시 터치가 된 느낌이 잘 나도록 효과를 주었다.

4. 개발결과 - 이미지 변환 API & 구글 AR Core 기능

4.1 이미지 변환 API



본 어플리케이션의 핵심인 실시간 사진 변환 서비스에 어떤 API, 어떤 기술을 적용해야 하나 상당히 고민하였다. 적용을 위해서 StarGAN, FaceApp API, 기타 Snapchat과 유사한 여러 어플리케이션의 기능 및

코드를 참조하려고 하였으나, StarGAN의 경우 정적인(static) 이미지에 대해서만 Pre-trained 학습 결과물을 나타내어서 적용이 어려웠다. 목표로 하는 어플리케이션은 카메라를 통해 얼굴을 인식하여 실시간으로 효과가 사진 위에 적용이 되는 AR 기능이어야 하였기 때문에 StarGAN을 채택하지 못하였다. 한편, FaceApp API나 기타 다른 시중의 앱들은 API 등을 적용하여 사용하는 것이 서비스 약관 (Terms of Service) 위반에 해당되었다. 실제로 FaceApp API의 제작자는 제작사로부터 API 제공을 중단하도록 요구받았다는 것이 확인되었다. 이는 요즘의 Snapchat이나 SNOW와 같은 어플리케이션의 인기를 생각해보면 당연한 것이라 사료된다. 현재 시장에서 상업적으로 굉장히 성공한 어플리케이션의 사례이기 때문에 소스코드 공개나, API 제공 등에 있어서 굉장히 철저하게 관리하고 있는 것 같다.

4.2 Google AR Core



결국, 본 어플리케이션에는 공개되고, 합법적인 어플리케이션 구현 방법인 Google ARcore를 사용하게 되었다. Google AR Core는 구글에서 제공하는 증강 현실 어플리케이션을 빌드하기 위해 제공하는 개발 키트이다. Google AR core는 이미 잘 알려진 얼굴 인식 기능 등 여러 가지 기능을 제공하며, 이러한 기능은 본 어플리케이션의 기능인 얼굴 인식 후에 얼굴 위에 AR 효과를 제공하는 데 있어서 유용한 기능이다.



Fox Effect
(by Google ARcore)

Google ARcore를 이용하여, 본 어플리케이션에서는 사람 얼굴의 귀, 눈, 코, 입 등을 감지하여 좌표를 알아 낸 후, 그 위에 여우 이미지 png 파일을 overlay하여 실시간으로 효과를 제공 하였다. 만약, 얼굴 좌표를 다르게 이용하고, 또 다른 이미지 파일을 사용 한다면 무수히 많은 효과를 제공할 수 있을 것이다.

5. 역할 배분

강동민	이미지 호스팅 API back-end 개발 전반, 전체 개발 총괄
안준수	Android front-end 개발 전반, 호환성 검증
정지훈	머신러닝 API back-end 개발 전반, 서버 개발

6. 개발 일정

	1 주차	2 주차	3 주차	4 주차	5 주차	6 주차	7 주차	8 주차	9 주차	10 주차
이미지 호스팅 API										
머신러닝 프레임워크										
안드로이드 앱 개발										
앱 연동 및 동작 확인										