



국민대학교
전자정보통신대학
컴퓨터공학부

캡스톤 디자인 I

종합설계 프로젝트

프로젝트 명	코웨이 (Coding Wherever Easily)
팀 명	Coway
문서 제목	결과보고서

Version	1.3
Date	2018-MAY-27

팀원	신 종혁 (조장)
	박 세용
	이 성준
	최 인정

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12


CONFIDENTIALITY/SECURITY WARNING

이 문서에 포함되어 있는 정보는 국민대학교 전자정보통신대학 컴퓨터공학부 및 컴퓨터공학부 개설 교과목 캡스톤 디자인 수강 학생 중 프로젝트 "코웨이 (Coway Wherever Easily)"를 수행하는 팀 "코웨이"의 팀원들의 자산입니다. 국민대학교 컴퓨터공학부 및 팀 "코웨이"의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

문서 정보 / 수정 내역

Filename	최종보고서-코웨이.doc
원안작성자	신중혁, 박세용, 이성준, 최인정
수정작업자	신중혁, 박세용, 이성준, 최인정

수정날짜	대표수정자	Revision	추가/수정 항목	내 용
2018-05-12	신중혁	1.0	최초 작성	최초 작성
2018-05-14	박세용	1.1	개발 내용 추가	2.2.1 개발 내용 추가
2018-05-15	최인정	1.2	개발 내용 추가	2.2.1, 4.1 개발 내용 추가
2018-05-23	이성준	1.3	추가 및 정리	개발 내용 추가 및 정리

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

목 차

1	개요	4
1.1	프로젝트 개요	4
1.2	추진 배경 및 필요성	5
2	개발 내용 및 결과물	10
2.1	목표	10
2.2	연구/개발 내용 및 결과물	12
2.2.1	연구/개발 내용	12
2.2.2	시스템 기능 요구사항	19
2.2.3	시스템 비기능(품질) 요구사항	20
2.2.4	시스템 구조 및 설계도	21
2.2.5	활용/개발된 기술	22
2.2.6	현실적 제한 요소 및 그 해결 방안	22
2.2.7	결과물 목록	23
2.3	기대효과 및 활용방안	24
3	자기평가	25
4	부록	25
4.1	사용자 매뉴얼	25
4.2	테스트 케이스	27
4.3	기술 문서	28

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

1 개요

1.1 프로젝트 개요

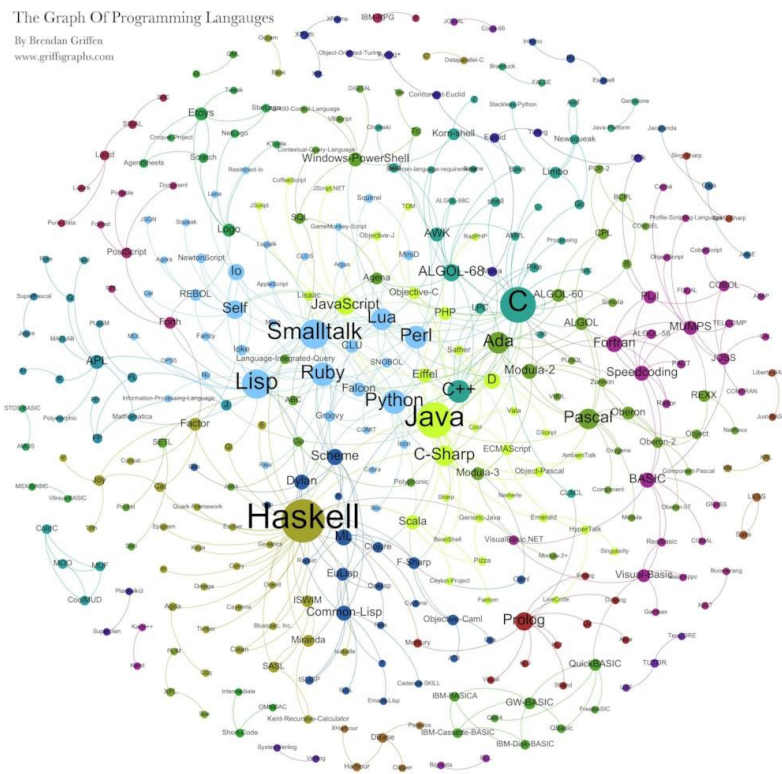


본 프로젝트(이하 Coway)는 **손으로 작성된 수도 코드**를 사진 촬영 또는 기존에 저장된 코드 이미지를 전송하여 Image Processing 과정 및 Interpreting 과정을 통해 텍스트 형태의 **완성된 Java 코드로** 변환해주는 **Android Application 개발 프로젝트**입니다



1.2 추진 배경 및 필요성

1.2.1 추진 배경



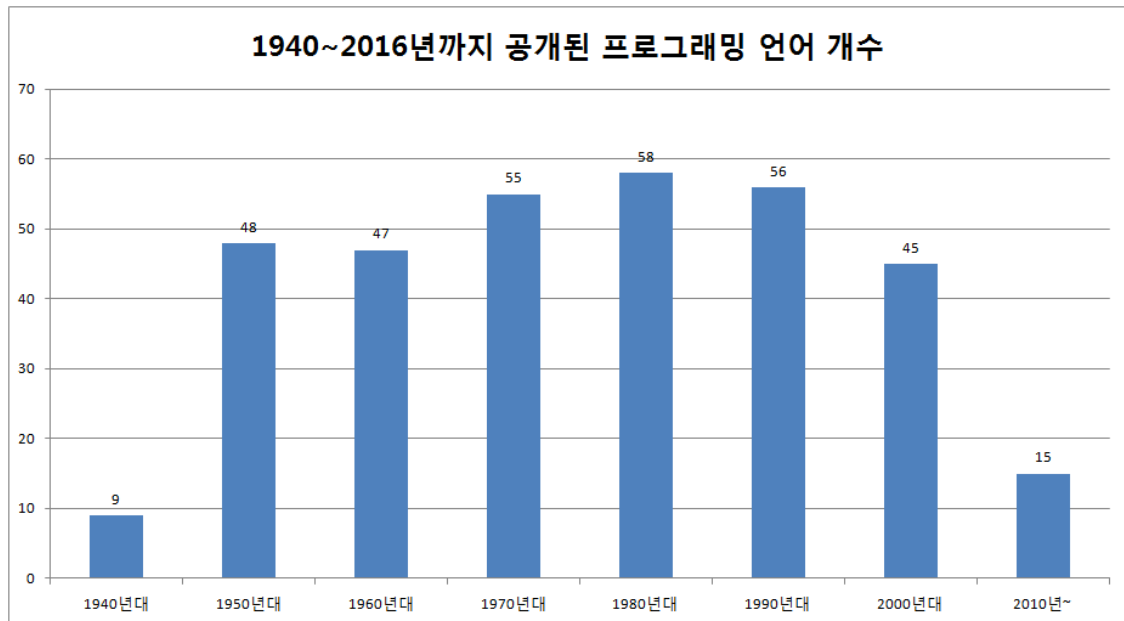
[출처: 나무위키 - 프로그래밍 언어/종류]

(<https://namu.wiki/w/%ED%94%84%EB%A1%9C%EA%B7%B8%EB%9E%98%EB%B0%8D%20%EC%96%B8%EC%96%B4/%EC%A2%85%EB%A5%98>)

IT 분야의 발달과 함께 많은 프로그래밍 언어가 생겨나고 활용되고 있다. 2000년대 초 C/C++, JAVA 가 대중적인 인기를 누리고 있었고, 2006년 이후 웹에 대한 수요가 급격하게 늘어나면서 웹프로그래밍 언어들의 인기가 폭발적으로 늘어났다. 2010년 이후 스마트폰의 등장으로 모바일 프로그래밍이 대두되면서 안드로이드에 활용되는 JAVA나 C# 뿐만 아니라 신생 언어인 Kotlin, Swift 등이 큰 성장세를 보였다. 최근에는 머신 러닝이 주목받으면서 데이터 과학분야에 많이 활용되는 R, python 등의 인기가 높아지고 있으며, JavaScript와 Node.js와 같은 JavaScript 관련 프레임워크들의 인기 또한 눈부시다.

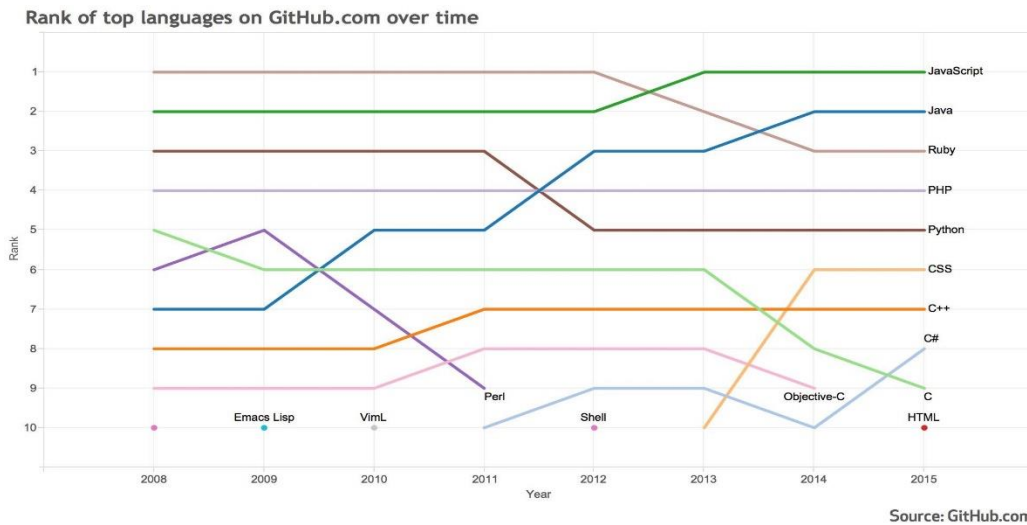


[출처: 위키피디아 - https://en.wikipedia.org/wiki/Timeline_of_programming_languages]



이와 같이 사회적 관심사와 수요에 따라 사람들이 관심을 가지는 언어가 다양하게 변화해왔으며, 그에 따라 IT 분야는 과거에 비해 다양한 언어들의 사용이 혼잡 되어있는 형태를 가지고 있다. 위키피디아에서 제공해주는 위의 표를 보면, 과거에 비해 새로 출시되는 언어의 수는 현저하게 줄고 있는 추세이다. 하지만 대기업들이 시장을 장악하기 위한 수단으로 새로운 언어를 출시하고 있어 시장 내 언어의 변화는 여전히 지속적으로 발생하고 있다.

하지만 각기 다른 언어들은 각자 상이한 문법을 가지고 있다는 점에서 소통의 문제점을 가지고 있다. 크게 비교해 보았을 때 C와 같이, 과거에 주로 사용되었던 언어들이 상대적으로 엄격한 문법을 가지고 있는 반면 최근에 인기를 얻고 있는 python과 같은 언어들은 직관적이고 비교적 자유로운 형태를 가지고 있다. 따라서 같은 코드를 보더라도 사용된 언어에 따라 코드의 이해도가 달라지는 문제점이 야기되고 있다. 이러한 문제점은 각기 다른 언어를 주 언어로 사용하고 있는 사용자간의 협업 시 의사소통을 방해하는 요소로 작용하고 있다.



[출처: <https://github.com/blog/2047-language-trends-on-github>]


이러한 문제점을 해결하기 위하여 문법을 최소화한 형태인 수도 코드를 활용하여 협업자들간의 의사소통을 원활하게 하되, 그 과정에서 발생하는 수도 코드를 컴퓨터가 이해 할 수 있는 상위 언어 코드로 자동 변환해주어 사용성을 높이면 어떨까 라는 생각으로 해당 프로젝트를 추진하게 되었다.

또한, 수도 코드를 사용하는 다수의 경우 컴퓨터가 아닌 수기로 작성한다는 점을 고려하여 프로젝트를 확장시켰다. 이에 따라 수기로 작성된 코드를 이미지 추출을 통해 텍스트로 변환해주는 기능을 추가한 코드 변환기를 프로젝트 목표로 설정하였다.

1.2.1.1 코드 변환기의 시장현황

현재 코드 변환기는 실질적인 사용량이 많지는 않으나 필요성에 의해 점차 다양한 제품이 생산되고 있는 분야 중 하나이다. IT 시장의 급속한 발달과 대응하여 다양한 언어들이 빠른 속도로 생겨나고 있으며, 각각의 언어는 언어의 특성을 살리기 위해 변형된 문법의 사용을 요구한다. 이에 따라 개발자들은 프로젝트에 사용되는 언어에 따라 소프트웨어 생산을 위한 시간이 아닌 단순한 언어 활용을 위한 별도의 시간을 투자해야 하는 문제에 직면하였고, 이러한 문제를 해결하기 위한 방안 중 하나로 코드 변환기가 대두되고 있다. 따라서 현재 프로그램 기반 혹은 웹 기반으로 언어 간의 변환을 지원하는 서비스들이 출시되고 있으며, 유료 콘텐츠 뿐 만 아니라 무료로 사용 가능한 서비스들이 점차 생겨나고 있어 이후 시장의 확장을 기대해 볼 수 있다.

하지만 현재 시중에 나와있는 코드 변환기의 경우 컴퓨터가 바로 이해할 수 있는 상위 언어간의 변환을 지원해주는 것으로, 수도 코드와 상위 언어 간의 변환을 지원하는 서비스는 출시되지 않은 상태이다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12


1.2.1.2 유사 서비스(상위 언어 변환기) 분석

1) 유사 서비스의 장점

	Service 1 (Program Based)	Service 2 (Web Based)
Common Positive	1. 연쇄성 - 변환 시 다양한 포맷의 상위 언어로 변환할 수 있으며 변환된 코드를 다시 원래의 언어 또는 다른 언어로 변환 가능 2. 명확성 - 문법이 명확하게 한정되어 있는 언어간의 변환으로 변환 시 높은 정확도 보장 3. 다양성 - 변환 가능한 언어의 종류가 많고 변환 코드의 포맷의 커스터마이징이 가능하여 다양한 사용자 요구 충족 가능	
Specific Positive	1. 파일 단위 및 폴더 단위로 변환 대상 지정 이 가능 2. 변환코드의 포맷(인덴트, 대괄호, 제어문 형태 등)의 다양한 설정 가능 (변환 만족도 향상)	1. 모바일을 활용하여 web app에 접근 가능 2. 변환 가능한 언어 목록을 쉽고 가시적으로 확인 가능

2) 유사 서비스의 단점

	Service 1 (Program Based)	Service 2 (Web Based)
Common Negative	1. 공유 기능 미흡 - 변환된 결과물이 프로그램 및 웹 내에 남는 방식만을 지원하여 사용자가 공유를 원할 경우, 그를 위한 부차적인 작업을 요구 2. 언어 문법에 대한 이해 필요 - 변환되는 각 언어간 특성 및 지원하는 문법을 이해하고 있어야 변환 가능 3. 비용적 부담 - 시중에 나와있는 제품의 대다수가 별도의 비용을 지불하는 방식으로 개발자들의 유입을 제한	
Specific Negative	1. 윈도우 기반 개발 환경에 접근해야 하므로 시공간적 제약 2. 각 변환언어마다 프로그램 설치 필요 (java->c#, c#->java 다른 2개의 프로그램 설치)	1. 간편성 위주로 코드 특유의 특성을 반영할 만한 설정 부재 (변환 정확도 및 만족도 하락) 2. 모바일을 활용한 접근 시 작성 속도 및 편의성 하락

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

1.2.2 필요성

1. 코드 작성의 시공간적 제약

- 출퇴근, 외출, 통학 시간과 같이 개발 환경에 접근하기 힘든 상황에서 고민하고 있던 해결해야 할 문제에 대한 아이디어가 갑작스럽게 떠올라도 아이디어 기록 및 코딩을 하기 힘든 상황이 발생하며 기타 저장 매체에 간단한 수도 코드 형식으로 기록이 되더라도 분실 혹은 수도 코드의 문법 미 적용 등의 문제로 실제 코드를 작성할 때 활용되기 힘든 일회성 정보로 사라지는 경우 발생

2. 수 많은 언어간 상이한 특징 및 문법, 프로젝트 의사소통의 비효율성

- 자바, Python, C 등 수 많은 언어마다 메모리 관리, 자료 구조, 변수 선언 등 접해보지 않았다면 이해하기 힘든 수 많은 서로 다른 특징 및 문법이 분명히 존재하고 있으며 공동으로 진행하는 프로젝트 특성상 서로 다른 언어, 플랫폼의 개발자 간(ios/android, java/python framework...) 공통된 로직을 구현하기 위해 의사소통을 하는 일이 상당히 많은데 서로의 코드로 작성된 아이디어로 의견을 공유하다 보면 개념에 앞서 각 언어간 차이점을 이해하기 위한 시간 낭비로 아이디어의 본질에서 멀어지는 경우가 발생

3. 코딩의 높은 진입 장벽

- 프로그래밍언어 비숙련자일 경우 비교적 언어 종속성에서 자유로운 알고리즘, 아이디어의 경우 고심을 통해 아이디어가 생각나더라도 이에 초점을 맞추고 발전 및 구현하기도 부족한 시간에 각 언어간 특징적인 문법적인 요소 고려 등의 문제로 더 많은 시간과 노력이 낭비되어 개발이 더뎠지는 상황이 발생

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

2 개발 내용 및 결과물

2.1 목표

손으로 작성된 수도 코드를 사진으로 찍어 텍스트 형태의 상위 언어 코드로 변환해주는 프로그램을 개발하는 것을 목표로 한다. 길지 않은 간단한 알고리즘을 표현 할 수 있는 정도의 문법을 변환 가능 범위로 설정하였으며, 주로 많이 사용되는 수도 코드 표현들을 취합하여 해당 표현들로 작성된 코드를 많이 쓰여지는 상위 언어인 자바 형태로 변환 해주는 것을 프로젝트 범위로 설정한다. 그 결과 코딩의 시공간적 제약을 줄이고 문법에 의한 언어의 진입장벽을 낮춰, 다양한 사용자들이 보다 알고리즘에 초점을 맞춰 의사소통 할 수 있는 환경을 제공한다. 이로써 문법에 익숙하지 않은 아이들을 위한 코딩 교육 도구, 서로 다른 언어를 사용하는 개발자들의 의사소통 방안 등으로 활용할 수 있을 것으로 기대한다.



1. Application(Front)

- **튜토리얼 진행** : 어플리케이션을 최초 실행 시 간단한 튜토리얼을 진행하여 COWAY Application 수도 코드의 문법을 스스로 작성하며 사용법을 익힐 수 있게 한다.

- **수도 코드 사진 촬영 및 전송** : 처음 구동이 아니라면 작성한 수도 코드를 사진 촬영 버튼을 눌러 사진을 촬영한 후, 사용자가 원하는 구역을 지정하는 등 촬영된 사진을 수정

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

하여 최종 완료 및 서버로 전송한다

- **저장된 수도 코드 사진 호출 및 전송** : 수도 코드의 사진 촬영이 아닌 경우 갤러리에서 불러오기를 통하여 이전에 작성한 수도 코드 사진을 불러온 후 서버로 전송한다


- **변환 완료 코드 수신** : 변환이 완료된 자바 코드는 다시 COWAY APP으로 전송되며 사용자가 작성한 수도 코드와 변환이 완료된 자바 코드를 상하로 배치하여 한눈에 확인할 수 있다. 사용자는 내용을 확인하여 변환이 정확할 경우 이를 스마트폰에 따로 저장하거나 이메일, SNS등의 경로로 결과 공유가 가능하다. 변환이 정확하지 않은 경우 다시 1번으로 돌아갈 수 있다. 변환 자체가 이루어지지 않은 경우에도 초기화면으로 돌아갈 수 있으나, 변환 자체를 무시하고 남은 과정을 진행 할 수 있다

2. Server(Back-end)

- **텍스트 추출** : 서버는 사용자가 전송한 사진을 수신하면, 서버 내부의 텍스트 추출을 위한 영상 처리 모듈을 실행한다. 영상 처리 모듈은 내부 텍스트 추출 알고리즘을 활용하며, 기존에 구현한 알고리즘을 라이브러리로 불러와 추출을 실시하여 최종적으로 수도 코드 이미지 파일에서 수도 코드 텍스트 파일을 생성한다

- **수도코드 전처리** : 수기로 작성된 수도 코드는 작성한 사람의 각기 다른 필기체에 따라 글자의 명확성이 달라질 가능성이 존재하기 때문에 정확한 변환을 위해 전처리를 통하여 수도 코드를 정제한다. 추출된 텍스트를 자바로 작성된 전처리 프로그램을 실행시켜 각 라인마다 검사를 실시하고, 수정이 가능한 오류는 Java코드 변환 프로그램이 인식할 수 있는 수도 코드 문법으로 수정한다. 인식할 수 없는 문법이거나 혹은 텍스트가 잘못 추출되었을 경우 반환을 하지 않는다

- **수도 -> Java코드 변환** : 전처리를 완료한 수도 코드는 예약어 매칭 방식을 활용한 인터프리터 모듈에서 자바 코드로 변환이 완료된다. 인터프리터 모듈은 수도 코드를 한 문자씩 불러들여 해당 문자와 예약되어있는 자바 코드 매칭을 확인하여 변환을 실시한다. 이 작업은 라인 단위로 진행이 되며, 만약 매칭되지 않은 예약어가 발견되어 변환에 오류가 발생할 경우 해당 라인에 표시를 하고 변환을 멈춘다. 변환이 성공적으로 완료되면 완성된 자바 코드를 텍스트로 사용자에게 송신하게 되고, 문법적인 오류가 발생할 경우 예러가 발생한 수도 코드 라인을 표시하여 사용자에게 전송한다

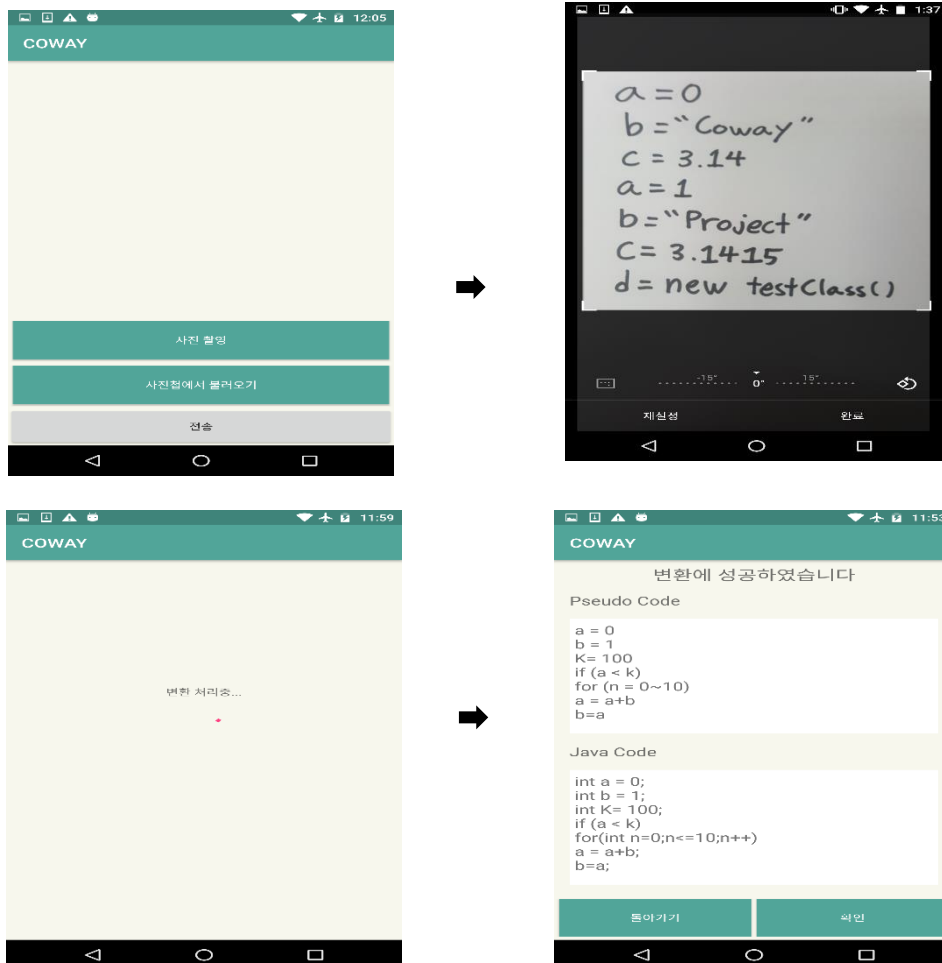
 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12


2.2 연구/개발 내용 및 결과물

2.2.1 연구/개발 내용

1. Application (Front)

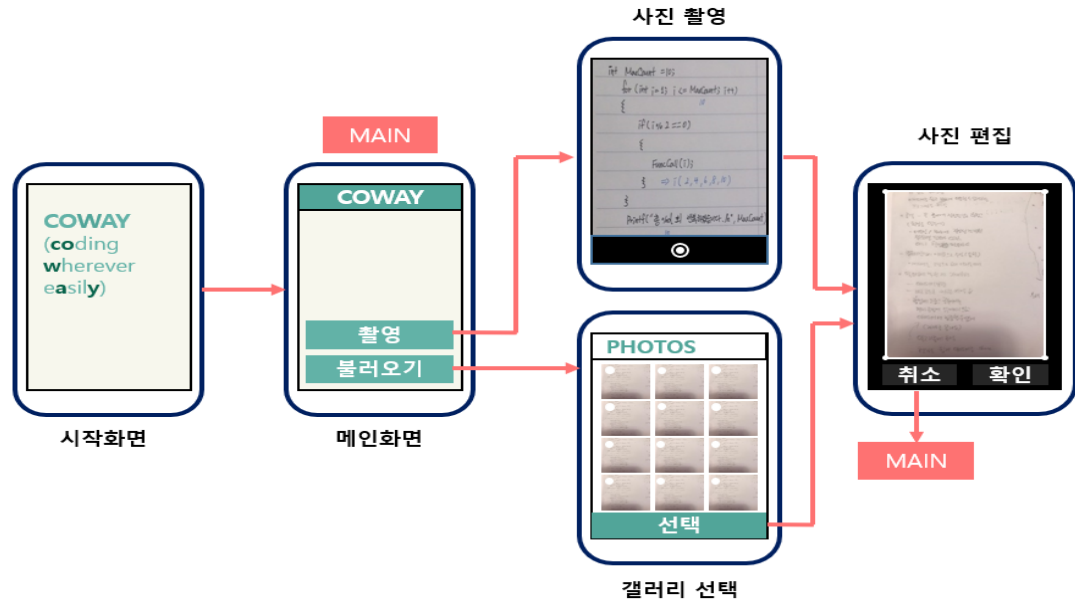
- **Commons Library** : 소켓 서버 및 각종 모듈들에 공통적으로 사용되는 기능들을 담아놓은 jar 라이브러리이다. 통신 프로토콜인 Packet 클래스부터, IO 모듈 등이 포함되어 있으며 모든 java 모듈들에서 의존하고 있다.
- **App 개발** : 사용자와 본 프로젝트가 직접적으로 상호작용하는 프론트로 Android Studio를 활용하였다. 변환할 대상이 될 코드를 촬영하거나 사진첩에서 사진을 불러오는 기능과 socket을 통해 서버로 사진을 전송하는 전송 기능, 변환된 코드를 사용자에게 보여주는 결과 화면과 결과물을 공유 및 저장하는 저장 기능으로 분리된다. 사진 선택 및 사이즈 조절을 위한 crop이 정상적으로 작동하고 있으며, 변환 결과를 세분화하여 그에 따른 결과 창을 분리해 사용자에게 안내하도록 제작되었다.



 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

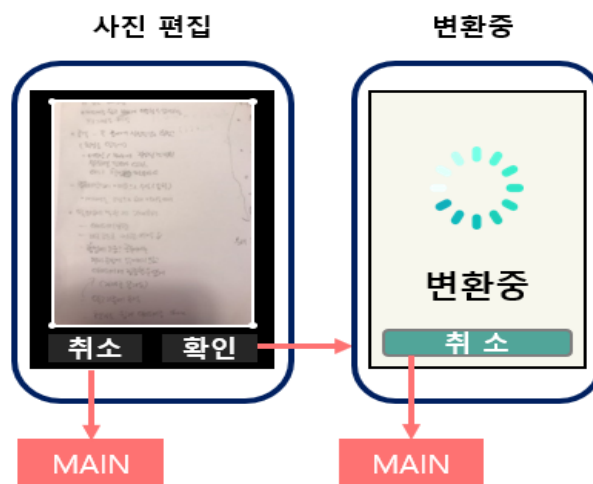
<세부 개발 내용>

● 사진 선택




- ① 사진 촬영
 - 촬영 후 사진은 갤러리에 저장, crop 기능으로 이동
- ② 갤러리에서 선택
 - 갤러리에 접근하여 사진 선택, crop 기능으로 이동

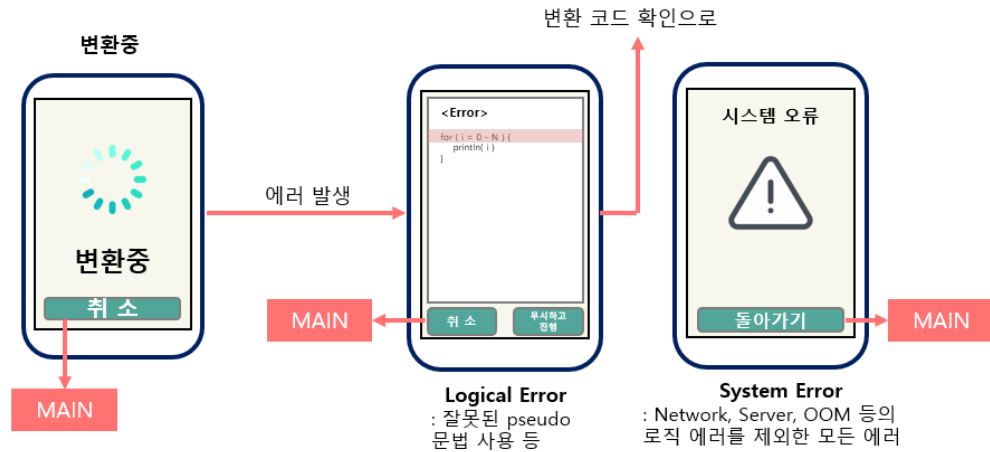
● 코드 변환 요청



- ① 사진 중 code 부분에 해당하는 영역을 crop
- ② 완료 버튼 클릭 후 상단 저장 버튼 클릭
 - crop 이미지는 갤러리에 저장되지 않고, 변환 처리 완료 후 소멸
- ③ 하단 전송 버튼 클릭 (Socket 통신을 통한 서버로의 이미지 전송)

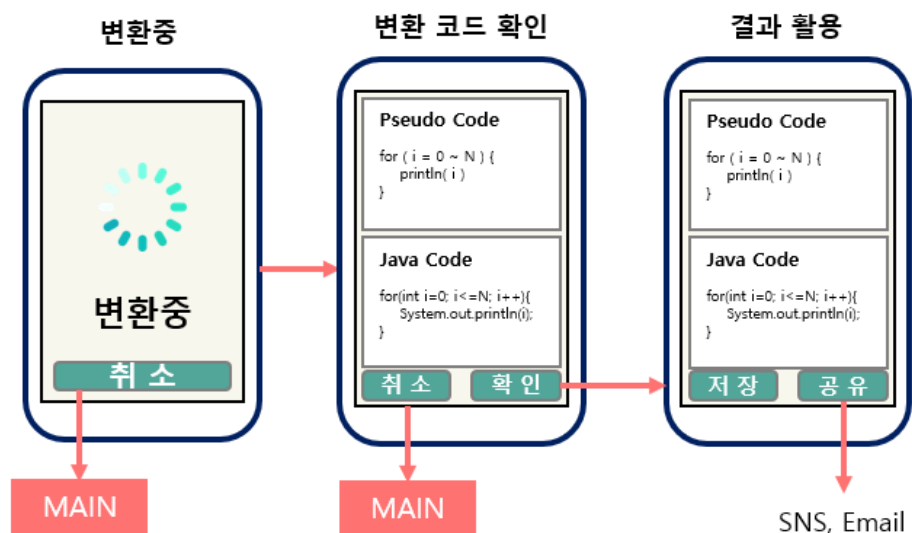
 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

● 코드 변환 실패 시



- ① Logical Error(영상처리 성공 & 수도 코드 변환 실패)
 - 서버로부터 수신한 socket의 status code가 200인 경우 발생
 - 변환되지 못한 라인 번호를 담고있는 array를 읽어 들여, 해당 번호 라인에 해당되는 코드를 하이라이팅해 사용자에게 변환되지 않음을 안내
- ② System Error(영상처리 실패)
 - 서버로부터 수신한 socket의 status code가 100/200이 아닌 경우 발생
 - 사용자에게 시스템적으로 오류가 발생함을 안내하고, 초기화면으로 돌아 갈 수 있도록 설정

● 코드 변환 성공 시



- ① 서버로부터 수신한 socket의 status code가 100인 경우 발생
- ② 기존 수도코드와 변환된 자바 코드의 비교 화면 출력

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

③ 공유 버튼 클릭 시 변환된 자바 코드 외부로 내보내기(e-mail)

2. Server (Back-end)

- **Java Socket Server 개발** : Coway의 pseudo code 변환 과정은 모두 서버에서 이뤄지기 때문에, Socket 통신을 처리할 소켓 서버가 필요하다. 프로젝트에서 사용할 모듈들의 대부분이 Java로 작성되어 있기 때문에, 소켓 서버도 호환을 고려하여 Java로 작성하여 jar파일로 배포한 뒤 구동한다.
- **Linux 서버 구축** : 영상처리 등의 무거운 연산을 수행하거나, 다양한 사용자의 요청을 처리하기 위해선 Coway 전용 서버가 필수적이다. 따라서 안정적인 운용을 위해 CentOS 7.3 기반의 Linux 서버를 구축하였다.

```
==== Client Info ====
Date: 2018-05-25_14:07:42
ip: /1.209.175.115

receivedImagePath: /root/project/data/img/img_1527224862570
pseudoFilePath: /root/project/data/pseudo/pseudo_1527224862592
```

- **Interpreter Module 개발** : Pseudo code를 .java 소스코드로 바꿔주는 변환 모듈이며 Java로 작성되었다. 변환 가능한 수도코드 형식들을 white list로 미리 등록한 뒤, 정규식 매칭을 이용해 변환을 시도한다. 변환 가능한 Pseudo code 내용으로는 for, println과 같은 메서드나, i = 10 과 같은 변수들이 있다. 성능 평가 결과로는 1000 줄의 코드에 대해 0.1초 내외로 변환이 가능하며, 정해진 화이트리스트의 코드들에 대해서는 모두 변환이 가능하다.



```
pseudoFilePath: /root/project/data/pseudo/pseudo_1527224862592
매칭 대상 라인: a = 0
[변수추론]
⇒ 좌변(a) 우변(0)rhs = 0      type = 1
⇒ 정규식으로 추론된 우변의 타입 (int)
⇒ 새로운 변수 (변수 선언 기록 테이블 = {a=int})
⇒ 매칭 성공
⇒ 매칭 종료 :: int a = 0;

매칭 대상 라인: b = 1
[변수추론]
⇒ 좌변(b) 우변(1)rhs = 1      type = 1
⇒ 정규식으로 추론된 우변의 타입 (int)
⇒ 새로운 변수 (변수 선언 기록 테이블 = {a=int, b=int})
⇒ 매칭 성공
⇒ 매칭 종료 :: int b = 1;

매칭 대상 라인: k= 100
[변수추론]
⇒ 좌변(k) 우변(100)rhs = 100  type = 1
⇒ 정규식으로 추론된 우변의 타입 (int)
⇒ 새로운 변수 (변수 선언 기록 테이블 = {a=int, b=int, k=int})
⇒ 매칭 성공
⇒ 매칭 종료 :: int k= 100;

매칭 대상 라인: if (a < k){
[예약어매칭]
⇒ 공백 제거: if(a<k){
⇒ 특수문자 기준 분리: [if, a, k]
⇒ 검사할 단어: if, 예약어 화이트리스트 = [print, println, for]
⇒ 검사할 단어: a, 예약어 화이트리스트 = [print, println, for]
⇒ 검사할 단어: k, 예약어 화이트리스트 = [print, println, for]
⇒ 예약어 매칭 종료

매칭 대상 라인: for (n = 0 ~10)
[예약어매칭]
⇒ 공백 제거: for(n=0~10)
⇒ 특수문자 기준 분리: [for, n, 0, 10]
⇒ 검사할 단어: for, 예약어 화이트리스트 = [print, println, for]
⇒ 예약어(for) 정규식 매칭 시작
⇒ 라인에서 변수들만 추출: [n, 0, 10]
⇒ 변수들을 자바코드 포맷에 채워서 완성: for(int n=0;n≤10;n++)
⇒ 예약어 매칭 종료
```




매칭 대상 라인: $a = a+b$

[변수추론]

⇒ 좌변(a) 우변(a+b) rhs = a+b type = -1

rhs = a+b type = -1

rhs = a+b type = -1

rhs = a+b type = -1

rhs = a+b type = -1

rhs = a+b type = -1

rhs = a+b type = -1

⇒ 추론 불가능한 타입

⇒ 매칭 종료 :: $a = a+b$;

매칭 대상 라인: $b=a$

[변수추론]

⇒ 좌변(b) 우변(a) rhs = a type = -1

rhs = a type = -1

rhs = a type = -1

rhs = a type = -1

rhs = a type = -1

rhs = a type = -1

rhs = a type = -1

⇒ 추론 불가능한 타입

⇒ 매칭 종료 :: $b=a$;

매칭 대상 라인:

[예약어매칭]

⇒ 공백 제거:

⇒ 특수문자 기준 분리: []

⇒ 검사할 단어: , 예약어 화이트리스트 = [print, println, for]

⇒ 예약어 매칭 종료

매칭 대상 라인: }


[예약어매칭]

⇒ 공백 제거: }

⇒ 특수문자 기준 분리: []

⇒ 예약어 매칭 종료

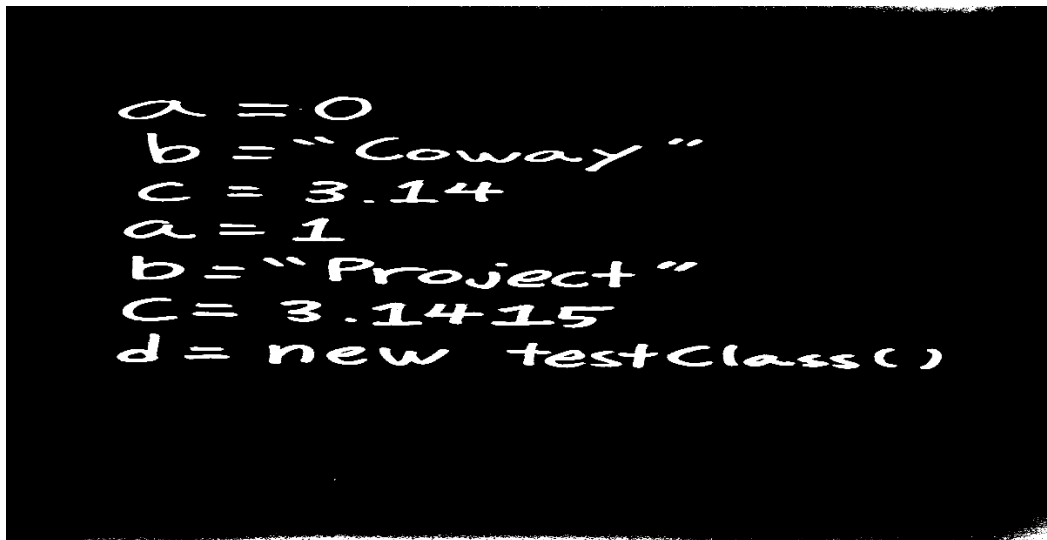
convertedFilePath: /root/project/data/converted/converted_1527224863649.java

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

- **외부 수도 코드 분석** : 프로젝트의 사용성을 높이기 위해 외부 수도 코드를 수집 및 분석하여 본 프로젝트에 반영하였다. 프로젝트의 범위에 해당되는 기호 기반 형태의 수도 코드를 수집하여 일반적으로 나타나는 기능과 표현식을 추출해 코드 변환 범위를 지속적으로 수정하였다.

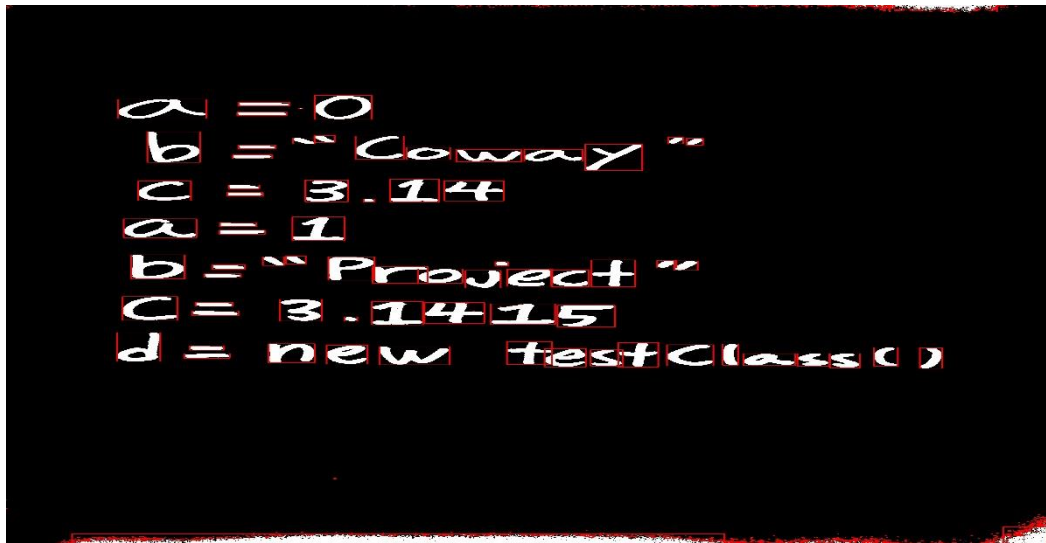
3. 영상처리

- **흑백 이미지 생성** : 텍스트 추출의 정확성을 높이기 위해 사용자가 전송한 이미지에서 색상을 제거하고, 이미지에서 명암을 없애는 작업이 필요하였다. 푸리에 변환을 통해 이미지에서 파장이 급격하게 변화하는 부분과 그렇지 않은 부분을 구별한 뒤, 임계값 변환을 활용하여 바이너리 이미지를 완성하였다.



- **이미지 contour 생성**: 바이너리 이미지 내부에서 각 글자들의 좌표를 설정하고, 분리되어 있는 i. / = / j 등을 하나의 글자로 인식하도록 하였다. 경계선 y축을 기준으로 일정 거리 이내에 존재하는 다른 contour의 x축 길이가 기준 contour의 x축 길이보다 작을 경우를 분리되어 있는 하나의 글자로 판단하였다. 또한 각 라인의 첫번째 글자의 y축 길이를 1.5배한 범위 이내에 존재하는 contour들은 같은 라인으로, 그 범위를 벗어나는 contour는 새로운 라인으로 구분하여 동적으로 생성되는 리스트에 라인별로 구별하였다.


 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12



- **Contour 클래스 생성** : contour 내부의 좌표, 각 축의 길이 등 접근해야 할 정보들이 많아 접근을 용이하게 하기 위한 contour 클래스를 별도로 제작하여 contour 관련 작업들은 객체 지향적으로 해결하였다.
- **피라미드 이미지 생성** : google vision api에서 인식이 용이하게 될 수 있도록 다양한 크기의 이미지를 픽셀 단위로 다수의 이미지를 생성, 피라미드 이미지를 생성하였다.
- **Google vision api** : 텍스트를 추출하기 위해 api 기반의 작업을 진행하였다.
- **텍스트 후처리** : 추출의 정확성을 높이기 위해 피라미드 이미지에서 추출된 각각의 텍스트를 종합, 분석하여 원본을 추론하고 원본 형태에 최대한 근접할 수 있도록 후처리를 실시하였다.

2.2.2 시스템 기능 요구사항

1. 사용자는 사진 촬영 혹은 사진첩에서 불러오는 것으로 수도 코드 이미지를 앱에 전송할 수 있어야 한다. (완료)
2. 사용자는 사진 크기 편집 기능을 통해 사진의 범위를 코드에 해당하는 부분만으로 한정시킬 수 있어야 한다. (완료)
3. 시스템은 변환 성공 여부를 사용자가 알 수 있도록 성공 여부를 포함하고 있는 결과 창을 사용자에게 제공해야 한다. (완료)
4. 변환 성공 시, 시스템은 사용자가 변환 전 후의 코드를 비교할 수 있도록 두 개의 뷰어 창을 제공해야 한다. (완료)

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

5. 변환 성공 시, 사용자는 변환된 자바 코드를 받는 방식(로컬에 저장하기, 메일로 전송하기, SNS 에 공유하기) 중 하나를 선택할 수 있어야 한다. (변경)

-변경사항: 프로젝트 개발 진행상황에 따라 일부 내보내기 방식이 제한 될 수 있음.

6. 변환 할 수 없는 수도 코드가 들어온 경우, 시스템은 변환 하지 못한 수도 코드를 표시한 별도의 에러 창을 사용자에게 제공해야 한다. (미완료)
7. 시스템 관련 에러가 발생한 경우, 시스템은 사용자에게 시스템 오류가 발생하였음을 안내하는 별도의 에러 창을 제공해야 한다. (완료)

2.2.3 시스템 비기능(품질) 요구사항

- A. **신뢰성** : 전체 중 코드 변환에서 에러가 발생하지 않은 횟수를 기준으로 80% 이상의 정확도를 가져야 한다.

- B. **효율성** :

- 앱에서 서버로 이미지를 전송하는데 소요되는 처리 시간은 2초 이내이어야 한다 (달성)
- 서버에서 앱으로 텍스트를 전송하는데 소요되는 처리 시간은 2초 이내이어야 한다.
- 이미지에서 텍스트를 추출하는데 소요되는 처리 시간은 3 초이어야 한다.
- 수도 코드에서 자바 코드로 변환하는 데에 소요되는 처리 시간은 5 초 이내이어야 한다

사용성 :

- 에러 발생 시, 에러의 종류를 명시하고 있는 별도의 창을 통해 사용자에게 안내해야 한다. (달성)
- 변환 처리 가능한 형식의 수도 코드 예제를 도움말과 사용자 튜토리얼로 제공해야 한다. (미달성)

원인: 아직 제작을 못함

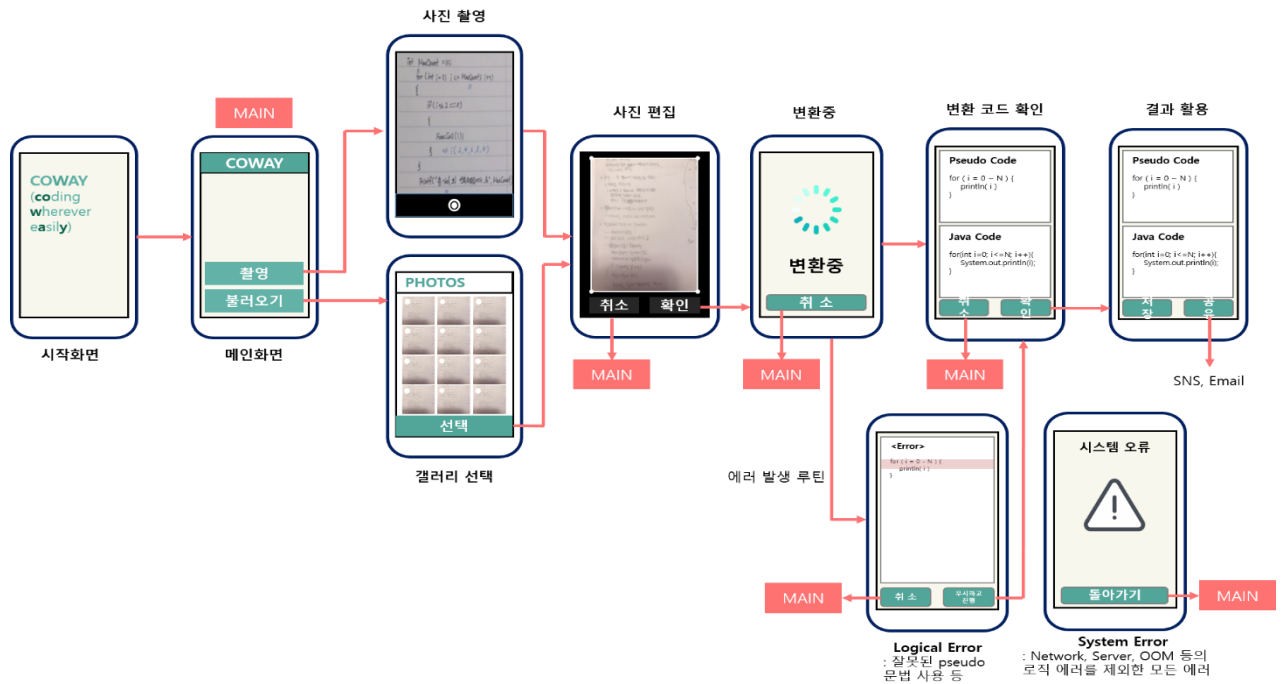
- 시스템은 안드로이드 6.0 이상의 모바일 환경에서 모두 작동되어야 한다. (미달성)

원인: 선택된 사진에 대한 crop 기능, 기기 내부 사진 접근 등의 앱에서 요구하는 기능들이 사용하는 기기의 제조사 별로 서로 상이하게 작동하고 있어, 기준으로 잡은 안드로이드 6.0 이상의 모든 기기들에 대한 테스트가 불가능하다고 판단. 따라서 Nexus 6p 모델을 기준으로 작동여부를 판단하는 것으로 프로그램 범위를 축소

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

2.2.4 시스템 구조 및 설계도

[Application UI]




[System Architecture]



● 변경사항

- 확장 모듈 제외: 프로젝트 주요 부분의 완성도를 높이기 위해 계획서에서 명시했던 확장 모듈을 해당 프로젝트의 범위에서 제외.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

2.2.5 활용/개발된 기술

- 1) Front
 - ① 카메라 및 갤러리 사진 연동 & crop
 - ② Socket을 활용한 io
- 2) back-end
 - ① Socket을 활용한 IO
 - ② Regex matching을 활용한 rule base Interpreting
- 3) Image Process
 - ① Python 기반의 openCV
 - ② Google Vision Api

2.2.6 현실적 제한 요소 및 그 해결 방안

1. OS 버전


해당 어플리케이션이 구동되는 안드로이드 기종의 OS버전이 6버전 전후로 많은 차이가 있어 개발 측면에서도 6버전 전후를 동시에 충족하는 프로그램을 개발하는데 한계가 있다. 이를 해결하는 방법으로 본 프로젝트에서는 안드로이드 6버전 이상의 기종을 대상으로 제한하였으며, 테스트의 기준으로 Nexus 6p 모델을 선택하여 해당 기기에서 동작 여부를 체크하며 개발을 진행하였다

2. 수도 코드의 비 정형성

특정한 포맷을 가지지 않고 작성되는 수도 코드의 특성에 따라 모든 수도 코드를 매칭시켜 자바 코드 형식으로 매칭시키는 것이 불가능하다. 따라서 현실적인 프로그램 진행을 위하여 본 프로젝트가 변환할 수 있는 수도 코드의 포맷을 일반화한다. 일반화 된 형태의 수도 코드는 외부 라이브러리에 관련된 기능을 제외하며, 순서도나 서술형 형태의 코드가 아닌 기호 기반의 수도 코드를 사용하는 것으로 범위를 설정하였다. 따라서 본 프로젝트에서 변환 가능한 수도 코드는 전체 수도 코드의 일부만을 대상으로 한다는 제한 요소를 가진다. 이 사항에 대해서는 1차 프로젝트 마감 후 사용자들이 작성한 수도 코드 데이터를 기반으로 점차 허용 가능한 수도 코드의 범위를 넓혀가는 형식으로 보완할 예정이다.

3. 수기 문자의 특성

일정한 굵기, 크기 등 문자의 특성이 일정한 폰트와 다르게 사람이 직접 작성하는 수기

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

문자의 경우 특성이 일정하지 않은 문제가 발생한다. 이로 인해 다음과 같은 문제가 발생하였다.

3.1 테저렉트 라이브러리 사용 불가

기존에 활용하고자 하였던 OCR 라이브러리 테저렉트는 폰트 기반으로 실행되는 라이브러리로 가벼우며 다양한 폰트를 지원하고 지원하지 않는 폰트의 경우 외부에서 1대1 매칭을 기반으로 학습을 시켜 성능을 개선 시킬 수 있는 장점이 있다. 그러나 수기의 특성과 학습을 시키기에도 현실적인 무리가 있었다. 사람의 글씨체는 매우 다르며 아무리 많은 사람의 수기 데이터를 학습하여도 불특정 다수의 사용자를 대상으로 하는 서비스의 특성상 통용될 만한 수준의 정확도를 테저렉트로 달성할 수 없었다.

3.2 Goolge Vision Api의 문제

머신 러닝을 기반으로 동작하는 구글 비전 api의 특성상 학습의 결과를 토대로 이미지에서 문자를 추출하는 방식으로, 매 학습의 결과가 이전 학습의 결과와 동일하다는 보장이 없는 것이 문제가 되었다. 특히 i/j, 0/o 등 특성이 유사한 문자에 대하여 이러한 문제가 두드러지는 경향을 보여 매 시도마다 서로 다른 결과를 보이는 경우가 잦았다. 명암, 글씨의 두께 등의 문제는 푸리에 변환 - 임계값 처리, 팽창/수축 변환으로 해결이 가능했지만 글씨체 등 글씨 외적인 문제에도 텍스트 인식의 결과가 크게 변화하는 문제가 발생하였다. 이러한 문제점들은 인식하기 용이하게 특성을 극대화한 글씨체를 사용하는 등의 방식으로 어느정도 해결할 수 있었으며, 이는 차후 서비스에서 튜토리얼 등으로 사용자에게 제공될 여지가 있다.

2.2.7 결과물 목록

대분류	소분류	기능	기술문서 유/무	비고
안드로이드	사진촬영	변환하고자 하는 코드를 이미지화	무	
	통신	이미지를 서버에 전송	무	
서버	영상처리	이미지에서 텍스트를 추출		

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

	<i>전처리</i>	수도 코드를 변환 가능한 형식으로 변형	무	
	<i>코드 변환</i>	수도 코드를 자바 코드로 변환	무	
	<i>통신</i>	앱에서 이미지 수신 및 변환 코드 전송	무	


2.3 기대효과 및 활용방안

2.3.1 기대효과

- 컴퓨터로 작성되지 않은 아이디어를 보관
- 즉각 코드로 변환해주어 코딩의 시간 및 공간적 제약 감소
- 문법에 얽매이지 않고 알고리즘에 초점을 맞출 수 있도록 지원
- 언어 별 상이한 문법에 따른 진입장벽을 감소

2.3.2 활용방안

- 1) 소프트웨어 교육에 활용
 - 문법적 제약이 적고 서술형 문장에서 변형된 형태인 수도 코드를 통한 접근으로 프로그래밍 언어의 진입장벽을 낮출 수 있음. 또한 앱을 활용하여 사진촬영 및 변환이라는 시각적으로 보여지는 프로그램 일련의 과정 참여를 통해 흥미를 유발시킬 수 있음
- 2) 알고리즘 공유 커뮤니티 활성화
 - 본 프로젝트를 확장하여 자바 만이 아닌 다른 언어들을 추가할 경우, 수도 코드가 언어들의 아우르는 하나의 기준으로 작용하여 서로 다른 언어를 사용하는 개발자들이 수도 코드를 통해 알고리즘에 대해 논의하며 의사소통 할 수 있음. 그를 통해 다양한 분야의 개발자들이 문법적 고민 없이 알고리즘 자체에 집중 할 수 있는 기반이 될 수 있음.
- 3) 다양한 언어로의 변환
 - 수도 코드 자체에 대한 의미 파악이 잘 이루어질 경우, 자바가 아닌 다른 상위 언어로의 변환이 가능. 이를 통해 하나의 수도 코드를 서로 다른 언어들로 변환하여 각 언어들이 형식 면에서 어떠한 차이를 보이는지 곧바로 비교 할 수 있음.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

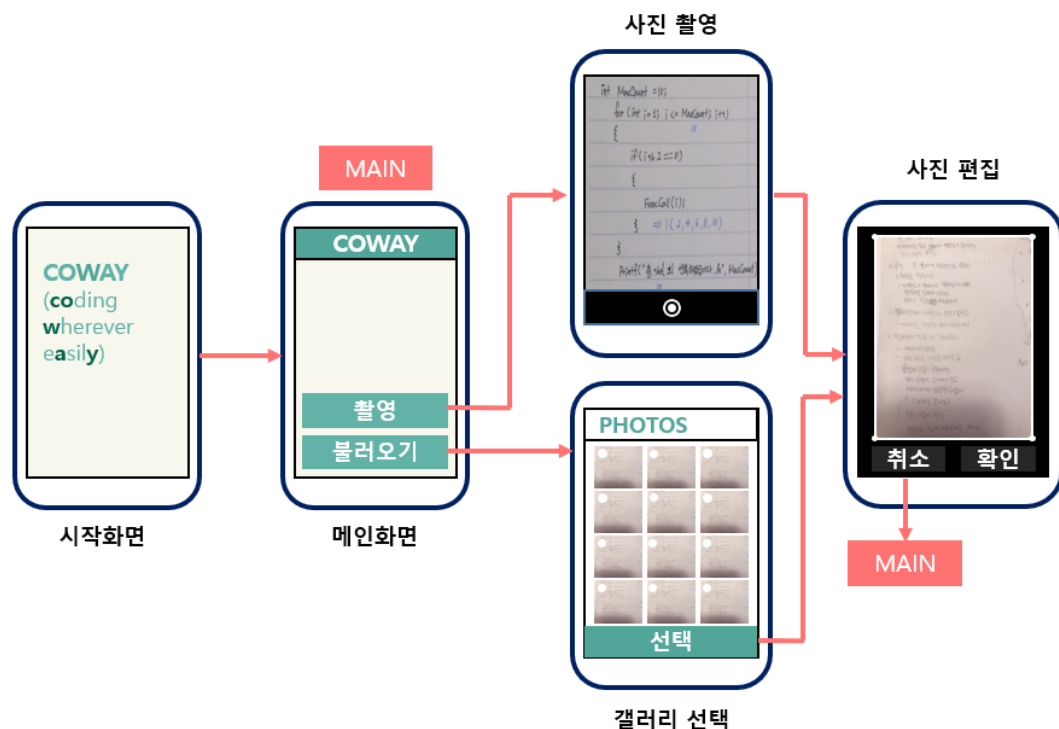
3 자기평가

- 초기 목표로 설정한 수도 코드에 비해 많이 구조화 된 형태의 수도 코드 인식만이 가능한 점이 아쉬움. 문법적 요소가 최소화된 수도 코드 변환을 통해 문법의 부담을 줄여 진입장벽을 낮추고자 하였으나, 머신 러닝을 포함하지 않은 현 프로젝트 상 순서도나 간단한 서술로 된 비형식적인 수도 코드 변환이 불가능한 한계. 축약된 수도 코드로 변환 범위를 축소시킨 결과 변환할 문법이 감소하여 현실성 있는 프로젝트 진행이 가능하였으나 그만큼 사용자에게 많은 문법적 규격을 요구한다는 점에서 아쉬움이 있음.

4 부록

4.1 사용자 매뉴얼

- 안드로이드 프론트 사진 선택



- ③ 안드로이드 카메라 촬영
- ④ 안드로이드 갤러리 사진첩에서 사진 선택
- ⑤ 중 하나 선택

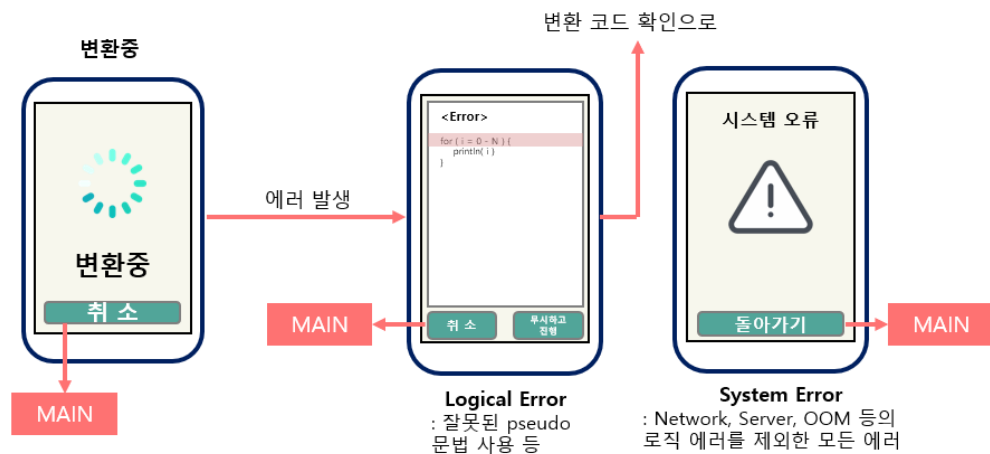
 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

● 안드로이드에서 코드 변환 요청




- ⑥ 사진 중 code 부분에 해당하는 영역을 crop
- ⑦ 완료 버튼 클릭 후 상단 저장 버튼 클릭
- ⑧ 하단 전송 버튼 클릭 (Socket 통신을 통한 서버로의 이미지 전송)

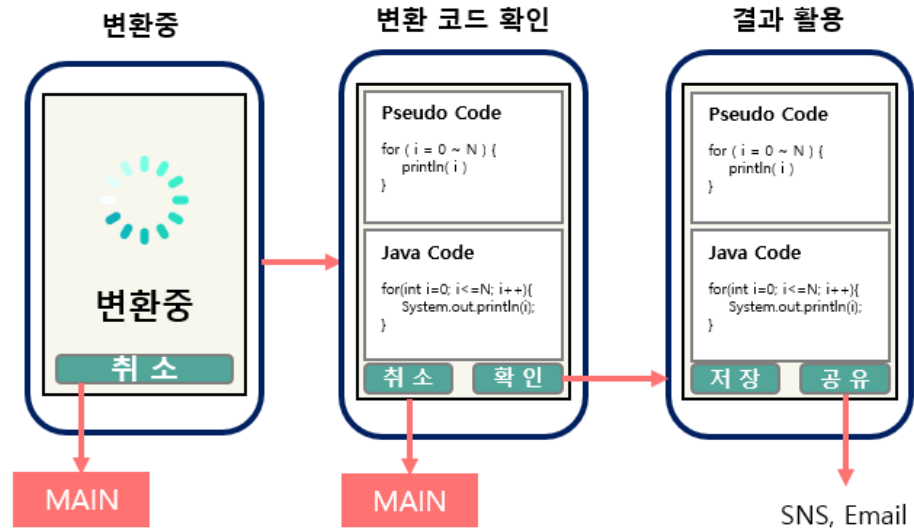
● 코드 변환 실패 시



- ③ 에러 종류별(Logical, System) 예외 표기되어 안내
- ④ Logical 에러 시 에러 발생 부분 하이라이팅

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12


● 코드 변환 성공 시



- ④ 기존 수도코드와 변환된 자바 코드의 비교 화면 출력
- ⑤ 공유 버튼 클릭 시 변환된 자바 코드 외부로 내보내기(e-mail)

4.2 테스트 케이스

대분류	소분류	기능	테스트 방법	기대 결과	테스트 결과
안드로이드	사진 선택	변환하고자 하는 코드 사진을 찍거나 갤러리에서 선택	<ul style="list-style-type: none"> - 촬영 버튼 클릭 시 카메라 기능 연동 여부 확인 및 crop 기능/저장 기능 동작 확인 - 갤러리 버튼 클릭 시 권한 문제 해당 기기 내부 사진 연동 여부 확인 및 crop 기능 동작 확인 	<ul style="list-style-type: none"> - 사진 촬영 시, 사진을 갤러리에 자동 저장 - 촬영 혹은 갤러리에서 선택 후 모두 crop 기능으로 연결되며, crop 후 변경된 사진은 저장되지 않는다. 	성공
	통신	이미지를 서버에 전송 및 결과 수신	-선택된 사진을 socket 을 통해 서버로 전송하고 전송되었는지 확인	-bytestream 형태로 전송된 사진이 서버 내 사진 디렉토리로 전송	성공

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

			-서버에서 결과를 socket 을 통해 전송할 때 앱이 이를 감지하고 결과에 맞는 화면창을 띄워주는지 확인		- 영상처리 실패 / 일부 변환 실패 / 성공 결과 여부에 따라 다른 화면창으로 연결	
서버	영상처리	이미지에서 텍스트를 추출	- 전송 받은 이미지를 전처리 하여 추출된 텍스트의 정확함을 확인	폰트 기반 수기 기반	- 사용자가 작성한 수도코드 원본 그대로 추출되어 저장된다.	성공
	전처리	수도 코드를 변환 가능한 형식으로 변형	추출된 텍스트를 코드 변환 모듈이 사용가능한 형식으로 수정한다.		코드 변환 모듈에서 사용가능한 형식으로 수정된다.	성공
	코드 변환	수도 코드를 자바 코드로 변환	추출된 수도코드 텍스트를 변환하여 자바 코드로 매칭한다.		동일한 의미의 자바 코드로 변경되어 저장된다.	성공
	통신	앱에서 이미지 수신 및 변환 코드 전송	-앱에서 사진을 socket 을 통해 서버로 전송했을 때 서버로 들어오는지 확인 -서버에서 결과를 socket 을 통해 전송할 때 앱으로 잘 전송되는지 확인		-사진이 서버 내 사진 디렉토리로 저장됨 - 변환 결과를 담은 socket 이 앱에 전송	성공


4.3 기술 문서

OpenCV 기초

- <https://opencv-python-tutroals.readthedocs.io/en/latest/>

푸리에 변환 - 1

- <http://blog.naver.com/PostView.nhn?blogId=samsjang&logNo=220565430664&parentCategoryNo=&categoryNo=66&viewDate=&isShowPopularPosts=false&from=postView>

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	코웨이(Coway Wherever Easily)	
	팀 명	코웨이	
	Confidential Restricted	Version 1.0	2018-MAY-12

푸리에 변환 - 2

●<http://blog.naver.com/PostView.nhn?blogId=samsjang&logNo=220568857153&parentCategoryNo=&categoryNo=66&viewDate=&isShowPopularPosts=false&from=postView>

윤곽선 생성

●<https://www.pyimagesearch.com/2014/04/21/building-pokedex-python-finding-game-boy-screen-step-4-6/>