

졸업작품 제안서 발표

2020.04.06 ~04.10

팀 구성

4

1팀 (3명)

작성글(0) 댓글(0)

입장

황보규(201212186)
(4학년)

박상우(201411271)
(4학년)

심준석(201611271)
(4학년)

15

2팀 (3명)

작성글(0) 댓글(0)

입장

서원준(201410349)
(4학년)

장승원(201411310)
(4학년)

이수민(201610099)
(4학년)

9

3팀 (3명)

작성글(0) 댓글(0)

입장

장혁준(201310805)
(4학년)

현인수(201512755)
(4학년)

조승현(201611300)
(4학년)

1

4팀 (3명)

작성글(0) 댓글(0)

입장

정준호(201113275)
(4학년)

전민규(201411802)
(4학년)

김도연(201614157)
(4학년)

5팀 (3명)

작성글(0) 댓글(0)

입장

강정모(201411257)
(4학년)

이한강(201411307)
(4학년)

이유진(201611284)
(4학년)

6팀 (3명)

작성글(0) 댓글(0)

입장

김민혁(201211330)
(4학년)

고길재(201410334)
(4학년)

조수빈(201510285)
(4학년)

7팀 (3명)

작성글(0) 댓글(0)

입장

서지혁(201313250)
(4학년)

강예나(201411255)
(4학년)

김나연(201610379)
(4학년)

8팀 (3명)

작성글(0) 댓글(0)

입장

정민원(201311313)
(4학년)

장원영(201411311)
(4학년)

함형준(201411318)
(4학년)

9팀 (2명)

작성글(0) 댓글(0)

입장

서지원(201411277)
(4학년)

홍나리(201511305)
(4학년)

5

14

3

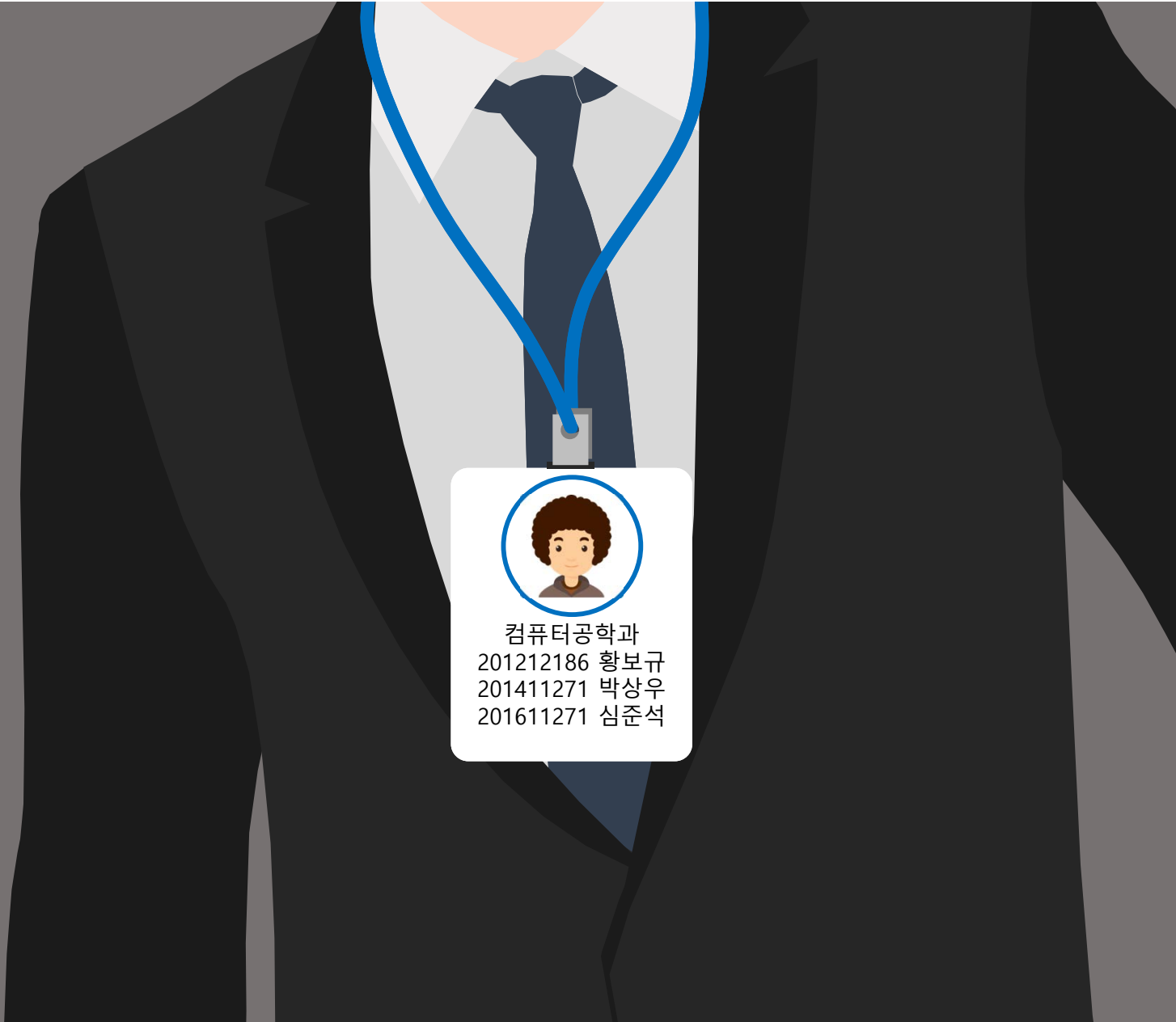
7

13

1조

프로젝트 제안서 1팀

저 여기서 내려요



저 여기서 내려요

지하철 탑승객 하차역 공유 서비스

Motivation

- 지하철에서 다른 승객들보다 먼저 승차했음에도 불구하고, 하차하는 승객 자리에 가까운 승객이 먼저 자리에 착석하는 경우가 빈번하게 발생. 먼저 지하철에 탑승했다면 자리에 먼저 앉을 수 있게 하여 편리성 증대.

Description (cont'd)

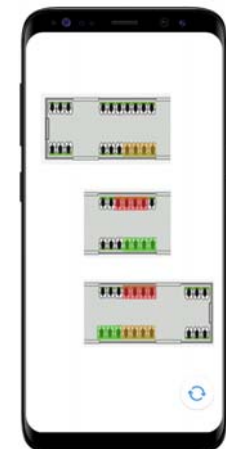
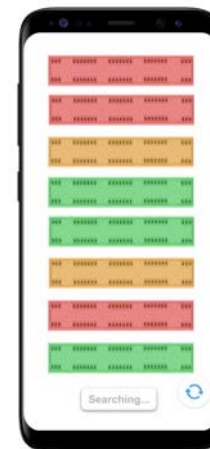
지하철 역 선택



노선에서 차량선택



차량내에서 객실선택/조회 객실내에서 하차역 등록/조회



하차역 조회/등록에
따라서 색깔 차이 有

저 여기서 내려요

지하철 탑승객 하차역 공유 서비스

Description

자신의 하차역을 등록했을 때에는 일정 포인트 지급, 다른 사용자들의 하차역을 조회할 때에는 일정 포인트를 차감하는 형식으로 하차정보를 조회하려는 사람만 발생하는 것을 막을 수 있다.

Review : 사생활 문제 , 제공되는 정보 실시간 변경

1. 사생활 문제

하차하는 역을 직접적으로 보여주는 방식 X -> 하차역이 3정거장 이내 : 빨강색

7정거장 이내 : 주황색

10정거장 이내 : 초록색 (정거장 수 기준은 변동가능.)

2. 제공되는 정보 실시간 변경

지하철 차량이 역을 지남에 따라 하차정보 변경 -> 새로고침 버튼을 통해 하차정보 다시 가져올 수 있음.

저 여기서 내려요

지하철 탑승객 하차역 공유 서비스

새로 만들 SW / COTS SW/ HW

- 저 여기서 내려요 Android App / 서울시 지하철 실시간 도착정보 API, Geolocation API , MariaDB / X

최종 산출물의 형태 및 기능



Android App : 사용자들간의 지하철 하차역 공유 서비스

Alternative Solutions

- 현재 live상태인 앱 서비스는 無
- 하차역을 표시할 수 있는 좌석을 개발, 지하철 차량내의 모든 좌석을 개발한 좌석으로 교체.

Project Justification

- 유사한 앱 서비스가 없으므로 시장을 선점할 수 있다.
- 모든 차량내의 H/W를 교체하는 것은 너무나 큰 비용이 든다.

저 여기서 내려요

지하철 탑승객 하차역 공유 서비스

Risk Analysis

- Common : 다수의 project들 동시 진행으로 인한 시간 부족.
코로나로 인한 수업 및 발표 일정 유동적이라서 혼선을 겪을 수 있음.
팀원들간의 면대면 소통이 어려움.
Open API 사용 경험 無
- Personal : Android Studio 사용 경험 無

Risk Reduction Plan

- 발표준비와 함께 설계 시작 (빠른 일정 시작)
실제 발표일정보다 3~4일 먼저 발표준비.
통학시간을 아껴 설계 및 개발에 힘쓰고, 팀원 톡방에서 진행상황 매일 공유.
팀원 1명이 발표준비를 할 때 다른 팀원 1명은 open API에 대한 활용법 조사 시작.
Android Studio 개발 경험이 있는 팀원 2명에게 도움을 받는다.

Success Criteria

- 차량내에 하차역을 등록한 승객들이 3명 있고, 1명이 하차역 조회시 누락없이 보여진다면 성공.
(팀원 3명 + 공기계 1개 총 4개의 기계를 기준으로 함.)

2조

← — □ ×

축소된 지식기반 데이터의 수집 Framework
이를 이용한 QA 인공지능 서비스

2팀_지식기반데이터_제안서

201410349	수학과	서원준
201411310	컴퓨터공학과	장승원
201410099	영어영문학과	이수민

0:00:01 0:03:19

🔊 💬 ⏮️ 10 ⏪ 30 ⏩ ⏭ 📄 ↗️ ⋮

2020-1_졸업프로젝트1

증강현실을 이용한 TCG 카드배틀

<TEAM>

201611300	컴퓨터공학과	조승현
201310805	기계공학과	장혁준
201512755	산업디자인학과	현인수

Motivation

- 기존의 TCG는 종이로 된 카드를 수집하는 재미가 있지만 **시각적인 재미가 부족하다.**
이를 **컴퓨터 게임**으로 만들 경우 시각적 재미는 있지만 실물이 없어 **수집하는 재미가 떨어진다.**

Purpose

- AR 기술로 **수집하는 재미**와 **시각적인 재미**를 모두 챙길 수 있는 게임을 개발하고자 한다.

Reference



■ DINOCORE PINI

다이너 피니는 2017년 11월 30일 출시한 스마트폰 AR카드게임이다.
다이너코어TCG의 경우 스마트폰의 카메라를 통해 인식한다.
플레이어들이 패의 카드와 스마트폰을 동시에 들고 게임을 진행하려면
불편하기에 본 프로젝트의 경우 고정된 카메라와 모니터를 이용한다.
<https://www.youtube.com/watch?v=Vj2IRcQCaac>

Project Definition

- 본 프로젝트는 종이로 된 카드를 이용해 게임을 진행하는 **카드배틀** 이다
- 게임은 모니터와 카메라가 세팅된 테이블 위에서 **3-4명의플레이어**가 진행한다.
- 플레이어가 카드를 내면, 해당 카드의 정보는 **카메라를 이용해 인식**한다.
- 모니터에서는 **플레이어들의 라이프**와 인식된 **카드의 3D 오브젝트** 그래픽을 보여준다.
- 게임 시작을 위한 세팅 이후 카메라는 플레이어들이 내는 카드를 계속해 인식하며 카드 간 상호작용이 자동으로 진행된다.

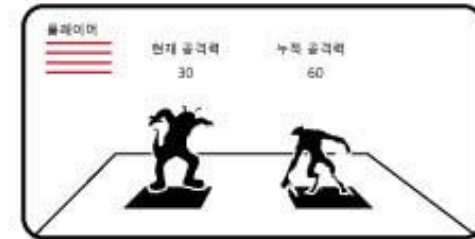


그림1. 예상되는 모니터 청사진 예시

COTS

- Camera API
- OpenCV

개발내용

- 카드의 정보를 담기 위한 이미지형태의 코드 및 암호화/복호화 하기 위한 알고리즘.
- 게임을 진행하기 위한 게임 알고리즘

Risk Analysis

- 종이로 된 카드이기 때문에 마모/파손됨에 따라 이미지코드가 훼손되어 인식이 안될 수 있다
- 카드게임 시 카드가 바로바로 인식되어 원활하고 빠르게 게임이 진행되어야 한다
- 카메라가 캡처하는 영상화면 중 이미지코드 부분을 찾아내야 한다.

Risk Reduction Plan

- 이미지 코드 내에 Error recovery 포함 시킨다.
- 코드 알고리즘을 최대한 간결하게 만들어 인식이 빠르고 오류가 적도록 한다.
- 이미지 코드를 특별한 색상을 이용해 카메라가 해당 부분을 바로 찾을 수 있도록 한다

Success Criteria

- 카드 인식이 3초 안에 이루어진다.
- 게임 진행 중 오류가 발생해선 안된다.

Schedule

	4월				5월				6월
	1	2	3	4(4/30)	1	2	3	4(5/31)	
프로젝트 제안서 발표(0407/0409)									
기능 설계									
이미지코드 설계									
게임알고리즘 설계									
중간보고서 발표(0428/0430)									
이미지코드 알고리즘 구현									
게임알고리즘 구현									
Interface 구현									
Debugging									
최종보고서 발표(0526/0528)									
전시회 준비(0602)									

4조

시각 장애인을 위한 상황 묘사 어플리케이션

Team 4

정준호(201113275)

전민규(201411802)

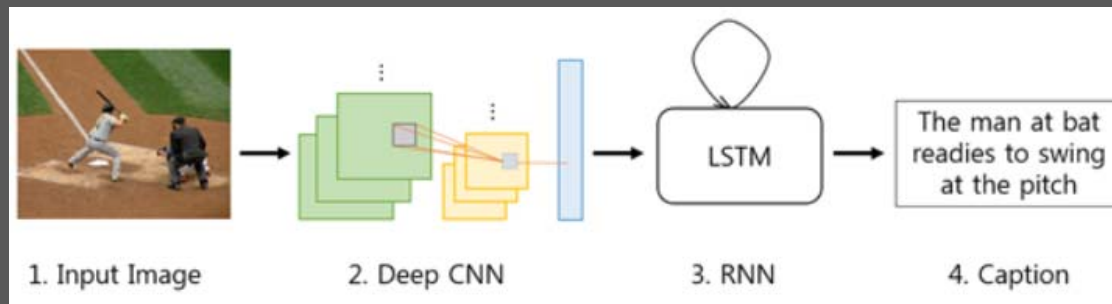
김도연(201614157)

'시각 장애인을 위한 상황 묘사 어플리케이션' 이란?

- 시각 장애인분들을 위해 카메라로 찍히는 사진의 상황을 글 및 소리로 묘사
- 시각 장애인들이 스마트폰을 달고 걸어가면 30초마다 한 장씩 사진을 찍어 그에 대한 설명을 음성으로 들려줌
- Model
 - Input : 사진 1장
 - output : 사진의 내용을 설명하는 단어(및 형태소)들이 나오고, 그 단어(및 형태소)들을 이어 붙여 문장을 만듦

$$y = y_1, \dots, y_C, y_i \in \mathbb{R}^K$$

Y : caption, K : vocab의 size, C : caption의 길이



※출처: <https://brain-kobli.co.kr/research.html>

새로 만들 SW

- 이미지 캡셔닝 모델 구현
이미지를 처리하는 CNN과 sequential한 정보를 생성할 LSTM을 seq2seq 구조로 이어 붙여 모델을 구현할 예정
- UI를 보여줄 앱

COTS SW

- TTS API

HW

- 스마트폰(보여줄 화면, 카메라, 스피커)

최종 산출물의 형태 및 기능

- APP
- 사진 한 장을 입력으로 넣었을 때 그 사진에 대한 설명이 텍스트 형태로 나오고 음성 메시지로 들려줌

Related Work

- Show attend and tell (이미지 캡셔닝의 기본 논문들)
- 기존 방법들은 방대한 영어 데이터를 바탕으로 학습된 모델이며 실제 한국어에 적합하게 만들어진 모델은 거의 없음(성능도 매우 안 좋음)
- 따라서 우리는 한국어를 위한 데이터를 새로 구축하고 한국어 특성에 맞게 학습을 하도록 함으로써 좋은 성능을 내는 모델을 구현할 예정임. 이는 한국어 자연어 처리 연구에도 큰 의미가 있음
- 아래의 실험은 약 10만개의 데이터를 번역하여 사용하였으며 우리는 더 적은 데이터로도 비슷하거나 더 좋은 성능을 내는 모델을 내는 것이 목표

표 1 학습데이터 유형별 실험결과

Model	B-1	B-2	B-3	B-4
어절 단위	0.289	0.190	0.141	0.111
의미형태소 단위	0.597	0.392	0.286	0.225
형태소 단위	0.615	0.430	0.322	0.251

Risk Analysis

- 데이터를 구축하는 데 어려움이 있음
존재하는 한국어 데이터가 없고, 방대한 영어 데이터를 직접 번역기로 번역하고 그것이 제대로 된 번역인지 일일이 체크해야 하는 노동이 들어감
- 시간 내에 성능을 올릴 수 있을지의 관건
현재 한국어를 위한 시도들은 있지만 Related work에서 보인 것 처럼 성능이 매우 좋지 않은 것을 알 수 있음.
- 딥러닝과 pytorch에 대한 이해도
정말 하고 싶은 주제지만 학부 수업에서 배울 기회가 없었기에 추가적인 공부가 필요

Success Criteria

- BLEU 0.6 이상 (더 적은 데이터(5만?)로 이전 방법과 비슷한 성능 기대)
- 30초 마다 한 번씩 찍고 10초마다 결과 나오도록

Another Photo Editor

5조

201411257 컴퓨터공학과 강정모
201411307 컴퓨터공학과 이한강
201611284 컴퓨터공학과 이유진

1. Outline – Another photo Editor (이미지 편집기)

- 포맷 변환, 대칭, 회전 기능을 갖는 이미지 편집 프로그램
- 자체적인 image DRM 기능 제공

※ 기능 세부

- 포맷 변환 : jpg ↔ png ↔ bmp 포맷 간 변환
- 대칭 : 상하좌우 대칭
- 회전 : 90°, 180°, 270°, 360° 회전
- 자체적인 image DRM 기능(Steganography 기법 활용)
 - Image 생성자가 고유의 자신의 signature image를 삽입하는 기능
 - Image 내의 삽입된 signature image를 확인할 수 있는 기능

※ 직접 구현 (python)

- 포맷 변환, 대칭, 회전, Image DRM 기능

※ open source

- Python image library
(디코딩 결과 나온 RGB 값들을 numpy array에 넣어 image 생성 후 화면에 출력하기 위해 사용)
- Scipy library
(DCT 등 압축 연산 시 수식 지원을 위해 사용)

2. 최종산출물 형태 및 기능

- 최종 산출물 : 이미지 편집 기능을 제공하는 SW
- UI: terminal에서 실행되는 text 기반의 UI

※ I/O

- input : 명령어 command, 편집 대상이 되는 이미지
- output : 결과 message, 편집 처리 된 이미지

- 1) 포맷 변환
 - 결과 message 출력 및 변환된 이미지 파일 속성 출력
- 2) 대칭, 회전, DRM
 - 성공 여부 출력 및 별도의 창을 띄워 변환된 이미지 출력

[포맷 변환]

<input> - 명령어 command, 편집 대상이 되는 이미지

```
jungmo_k@JungMo-K-MacBookPro ~ % transform ./aaa.jpg bbb.png
```

<output> - 결과 출력, message로 변환된 파일 속성 출력

```
bbb.png: PNG image data, 2658 x 1220, 8-bit/color RGBA, non-interlaced
jungmo_k@JungMo-K-MacBookPro Desktop %
```

[대칭, 회전, DRM]

<input> - 명령어 command, 편집 대상이 되는 이미지

```
jungmo_k@JungMo-K-MacBookPro ~ % rotate -r 90 ./aaa.jpg
```

<output> - 성공 여부 출력, 별도의 창을 띄워 변환된 이미지 출력

```
success
jungmo_k@JungMo-K-MacBookPro Desktop %
```

3. Alternative solution

- Image processing library : pillow, OpenCV, matplotlib
- 기존에 존재하는 SW

1) 알씨 : 알툴즈에서 지원하는 이미지 편집 GUI software.

장점 : 압축 파일 이미지 보기 및 다양한 포맷 간의 변환 지원

단점 : 딱히 없는 것 같음

2) 그림판 : 윈도우에서 기본적으로 지원하는 이미지 편집 GUI software.

장점 : 심플하고 가벼운 소프트웨어, 다양한 색상 선택 가능

단점 : 주로 그리기 기능 위주, 이미지 편집 기능은 기본적인 것들만 존재

3) image magic : 이미지 편집 기능을 제공하는 opensource software.

장점 : 이미지 합성, 암호화 및 복호화, openCL 알고리즘 지원, 패턴 인식 등 가장 많은 기능 지원

단점 : 2016년 이후 몇 차례 보안 취약점 문제 발생



대상 이미지



창으로 출력되는
변환된 이미지

4. Project justification

- 이미 비슷한 SW가 다수 존재하지만, 멀티미디어공학개론에서 배운 이미지 이론을 실제로 구현해보는 의의
- 그림판의 기본적인 이미지 편집 기능 + 알씨/image magic의 포맷 변환 기능
- 부가적으로 Steganography 기법을 활용한 디지털 워터마크 기술 사용으로 차별성 부여

5. Risk analysis & reduction plan

1) 포맷 변환 및 이미지 출력

- Jpg 및 Png의 인코딩 및 디코딩이 복잡함.
 - 시간 부족에 따른 risk 발생 가능.
 - ⇒ 다른 기능 축소 or 일부 API 사용
 - ⇒ 핵심 기능과, 기능의 중요도에 따른 구현 우선순위 부여
 - ⇒ 우선순위가 낮은 기능의 경우 부득이하게 API로 구현
 - 1) 포맷 변환(핵심) 및 Steganography 기반의 DRM 기술(차별성)
 - 2) 회전
 - 3) 대칭

2) 효율적인 steganography 기법의 구현

- 대부분 하위 bit에 data를 넣으나 압축과정에서 손실 발생
 - ⇒ 효율적인 steganography 기법에 대한 논문을 참고해야 함

※ References

- 멀티미디어공학개론 강의 자료 (bmp, jpg)
- 위키피디아 (png, steganography, DRM)
- File format wiki (png, jpg, bmp)
- Python 표준 library site
- Pillow document site (image library 사용)

3) 다양한 option 및 mode

- Bmp 및 jpg의 경우 header에서 다양한 mode를 지원함
 - ⇒ default mode에 대해서 먼저 개발

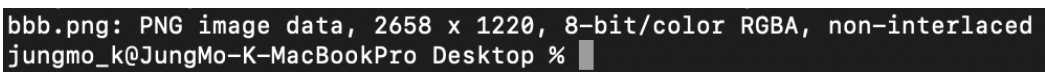
6. Success criteria

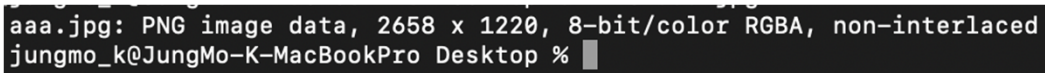
1) 완성도에 대한 기준

핵심 기능 : 포맷 변환, Steganography 디지털 워터마크 DRM 기술 구현
(전체 개발 기간의 80% 예측 – 80% 구현의 기준)

부가 기능 : 대칭, 회전 기능 구현
(전체 개발 기간의 20% 예측 – 20% 구현의 기준)

2) 각 기능의 구현 성공에 대한 기준

- 포맷 변환 : 파일 속성 output message로 확인
[제대로 성공한 경우, data format이 바르게 표시]


[단순히 파일 확장자 명만 바뀐 경우 data format에서 드러남]


- 대칭, 회전 : 변환 후 이미지를 별도의 창으로 띄워 실제로 확인
⇒ 입력한 조건과 동일하게 변했는지 직접 확인 가능

- Steganography 디지털 워터마크 DRM
⇒ Steganography의 경우 정보 은닉이 목적이므로
존재 여부는 따로 알려주지 않음.

※ 확인 절차

- 1) Steganography 기법으로 삽입할 image와 워터마크가 삽입될 image를 준비함
- 2) Steganography 기법으로 디지털 워터마크를 삽입
- 3) 다시 추출 했을 때 육안으로 동일하다고 식별 가능한 디지털 워터마크 image가 나와야 함

감염병 전파 예측 시스템

졸업프로젝트

고길재 | 김민혁 | 조수빈

01

프로젝트 개요

02

03

04

05

Description

확진자 자료를 이용해 감염병의 전파를 예측하는 웹 애플리케이션 WHO와 각 국가별 공식 사이트가 제공하는 데이터를 자동화된 크롤러로 수집하여 국가별 전파율 계산하여 감염병의 경과 예측 및 종료시기 제공

기능

- 감염병 전파 주요 국가 리스트 제공
- 국가별 확진자 수 제공
- 감염병 전파 주요 국가 이미지 제공
- 감염병 전파 모델 적용

SW / COTS SW

SEIR 모델
데이터 피팅 알고리즘

Graph
웹 크롤러
Python
Node.js, React
국가별 api

01

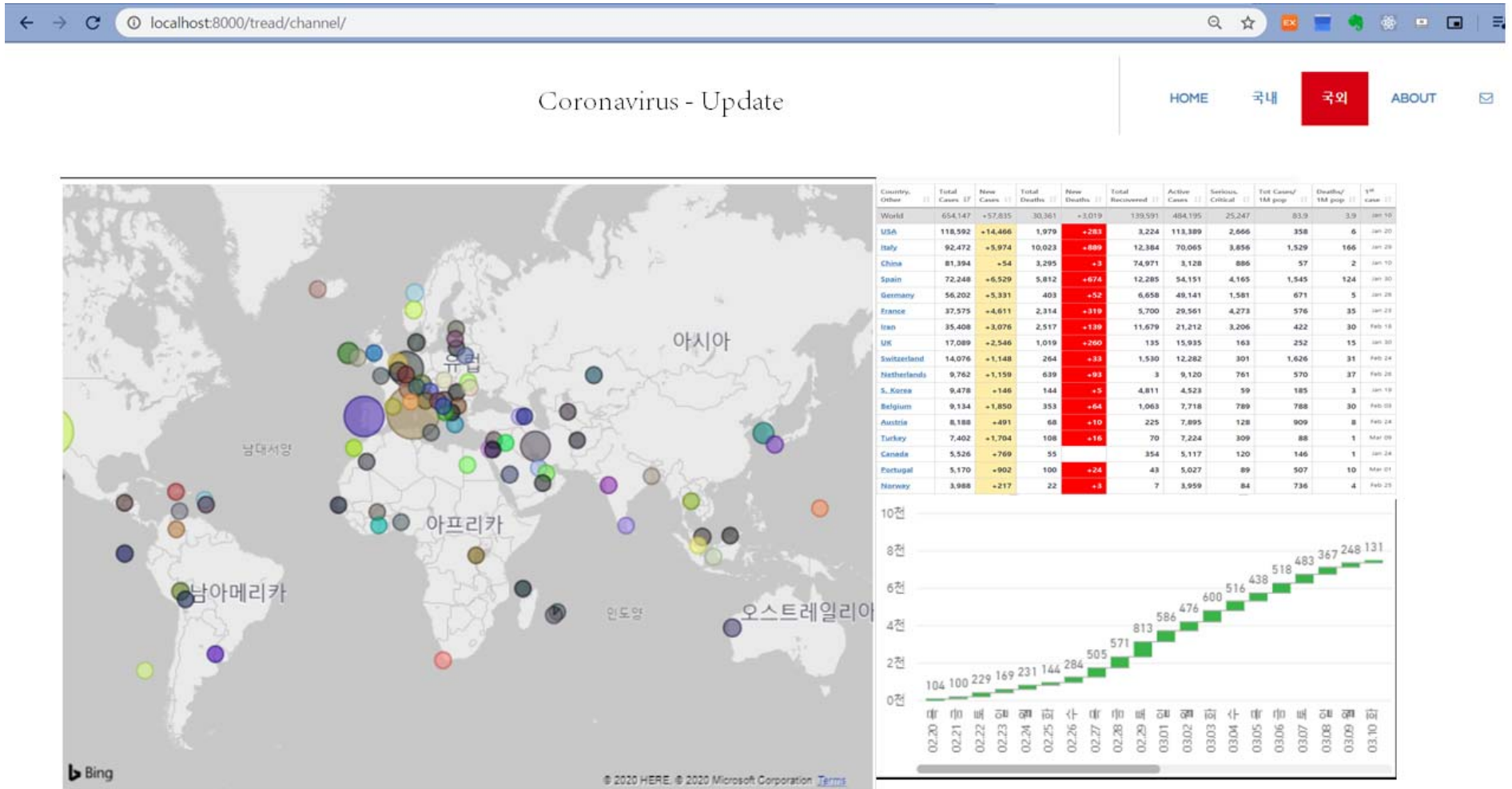
이미지 시안

02

03

04

05



01

기존 사이트와의 차별점

02

03

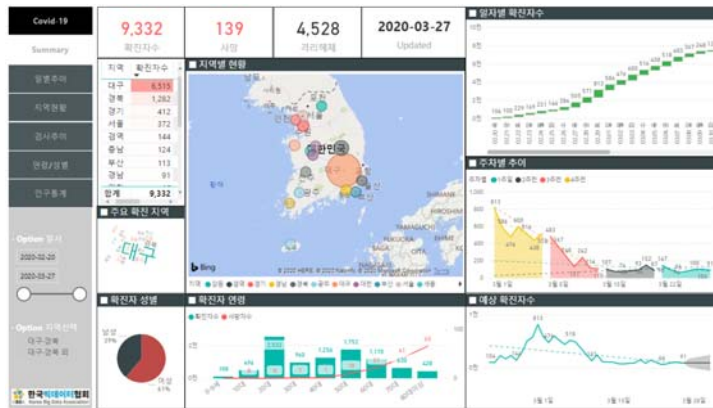
기존의 정보 제공 매체들은 사용자에게 전염병의 현황만을 제공한다는 한계
 사용자의 차후 계획을 위해서 전세계적인 예측 자료의 필요

04



05

수치적 기법으로 **감염병 전파정도를 예측**한 자료를 제공한다는 차별점



<https://k-corona.com>

Report coronavirus cases

Home Yesterday Search

Country/Officer	Total Cases	New Cases	Total Deaths	New Deaths	Total Recovered	Active Cases	Serious/Critical	Test Cases/1M pop	Test Deaths/1M pop
USA	93,151	+7,716	1,382	+87	2,424	89,345	2,432	281	4
China	81,340	+55	3,292	+5	74,588	3,460	1,034	57	2
Italy	80,589	8,215	8,215		10,361	62,013	3,612	1,333	136
Spain	64,059	+6,273	4,934	+949	9,357	49,768	4,165	1,370	106
Germany	49,344	+5,406	304	+37	5,673	43,367	23	589	4
Iran	32,932	+2,926	2,378	+544	11,133	18,821	2,893	385	28
France	29,155		1,696		4,948	22,511	3,375	447	26
UK	14,543	+2,885	759	+181	135	13,649	163	214	11
Switzerland	12,911	+500	207	+15	897	11,207	203	1,422	24
S. Korea	9,332	+91	139	+8	4,528	4,665	59	182	3
Netherlands	8,603	+1,172	546	+112	3	8,054	761	502	32
Austria	7,557	+648	58	+9	225	7,274	128	839	6
Belgium	7,284	+1,049	289	+69	858	6,137	690	628	25
Portugal	4,268	+724	76	+16	43	4,149	71	419	7
Canada	4,043		39		228	3,776	120	107	1
Norway	3,696	+324	19	+5	6	3,671	70	682	4
Turkey	3,629		75		26	3,528	136	43	0.9
Australia	3,180	+130	13		170	2,997	23	125	0.5

<https://www.worldometers.info/coronavirus/#countries>

01

02

03

04

05

Risk Analysis / Risk Reduction Plan

Success criteria

감염병에 관한 데이터가 한 곳에서 제공되지 않음.

➡ 각 국가별 공식 사이트를 통해 수집한다.

완치율(격리해제)에 대한 데이터가 없는 국가도 존재함

➡ 인접 국가들의 평균값 활용.

데이터피팅 후 함수의 결정계수 0.5이상을 목표로 한다.

지하철 Seat크릿

Project Name:

- 지하철 Seat크릿 (김나연, 강예나, 서지혁)

Description:

- '지하철 Seat크릿'은 기존의 지하철 노선도 어플에 실시간 좌석정보를 업데이트하여 지하철을 이용하는 사람들이 좌석대기를 보다 효율적으로 할 수 있도록 돕는 어플리케이션이다.
- 지하철 이용객이 좌석에 앉으면, 발 밑에 부착된 QR코드를 인식하여 도착역을 입력하고 도착예정시간을 알 수 있다.
- 좌석에 앉지 못하고 대기중인 승객은 QR코드 또는 직접 어플을 열어서 열차 칸의 공석과 좌석별 대기시간을 알 수 있다. 또한 동의 하에 공석이 생길 시 push알림을 받을 수 있다.
- 어플은 입력 받은 목적지와 도착예정정보를 서버에 업데이트한다.
- 좌석이 만석일 경우 이용객은 어플에 업데이트된 정보를 통해 효율적으로 대기할 좌석을 선택할 수 있다.

Deliverable:

- 좌석별 정보를 담고 있는 QR코드, 사용 가능한 어플리케이션



02 프로젝트에 필요한 것

지하철 Seat크릿

COTS SW:

- 서울 교통공사 지하철 API
- MySQL 서버
- zxing 라이브러리
- 노선도 그리기

SW(직접 개발할 부분):

- 지하철 현황, 앱 푸시, 사용자 여정 관리 등을 위한 데이터베이스 서버 구축
- QR코드 또는 직접입력을 통해 좌석정보를 열람하고 등록할 수 있는 안드로이드 어플리케이션

기타:

- 어플리케이션 ID와 열차, 좌석정보를 담고 있는 QR코드

지하철 Seat크릿

Alternative Solutions:

- '카카오맵' 하차알림 (여정의 끝에 푸시 알림을 보내주는 점을 참고)
- 서울 택시 안심귀가서비스 (QR 코드로 차량정보를 인식할 수 있는 점을 참고)
- '지하철 앉아가기': 서울시 과거 정보 공공데이터 API를 사용해 앉아서 갈 수 있는 확률을 알려주는 어플리케이션
- '지하철 빈 좌석 등록, 조회, 예약 시스템' 특허: 로그인 후 사용자가 직접 여정정보를 등록하고, 포인트를 받고 조회할 수 있는 어플리케이션
- '아두이노를 활용한 지하철 잔여 좌석 알림이'프로젝트: 아두이노 센서를 통해 공석 여부를 확인 후 표시한다.

Project Justification:

- 지하철의 경우 편리하게 좌석정보를 입력할 수 있는 방법이 없고, 지하철 좌석은 공유 자원임에도 불구하고 비효율적으로 활용되고 있다.
- 기존 어플리케이션 또는 프로젝트가 꾸준히 사용되지 않는 문제점은 대부분 편리성, 과정의 복잡성, 정보력 결여이다. 그에 비해 본 프로젝트는 QR코드를 사용한 직접적 접근으로 편리성을 높이고, 공석 여부나 확률에서 나아가 대략적인 대기시간을 제공함으로써 사용성을 높인다.

지하철 Seat크릿

Risk Analysis:

- 앱개발 경험이 없다.
- 서버응답시간이 매우 빨라야 효율적인 어플리케이션이다.
- 서울교통공사 지하철 API 장애 가능성 존재.
- QR코드를 인식하기 어려운 상황이 존재 할 수 있다.
- 거짓으로 도착정보를 입력하는 사용자 존재 가능성.

Risk Reduction Plan:

- 지속적인 안드로이드 프로그래밍 스터디를 통해 완성도 있는 프로젝트를 목표한다.
- 레이턴시를 낮출수 있는 방향의 프로그래밍.
- QR코드 인식 없이도 어플리케이션에서 현재시간과 출발역을 통해 간편하게 열차정보를 불러올 수 있도록 하는 기능 추가.
- 불일치하는 좌석정보 리포트 기능을 추가하여 조취한다.
- 장애 가능성에 대한 위험을 수용하되 본 앱을 최대한 api에 독립적으로 개발해 연관된 장애 가능성을 최소화한다



05 Success Criteria

지하철 Seat크릿

Success Criteria:

- 숙련된 사용자가 10초 이내에 QR 코드로 정확한 좌석 위치를 인식하고 여정 정보를 등록할 수 있다.
- 등록한 여정 구간의 좌석이 공석이 되었을 때 해당 위치를 5초 이내에 푸시 알림으로 받을 수 있다.
- 동시에 5개 이상의 입력이 들어올 경우 오류나 누락 없이 모든 정보를 정확히 표시할 수 있다.

8조

화면 분할 스트리밍

8팀 201311313 정인원

201411311 장원영

201411318 함형준

Overall

- 작품명 : 화면 분할 스트리밍
- 설명 : 3대의 스마트폰 화면을 연결하여 하나의 영상을 재생
- 만들 SW : Android App -> 네트워크 연결, 데이터 스트리밍, 화면 분할 제어
- 기존 SW : 비디오 재생 SW
- H W : 스마트폰
- 산출물 : 3대의 화면에 하나의 영상을 재생

Alternative Solution & Project Justification

- 다른 App에서는 3대의 기기가 모두 동영상 파일을 내장하고 있어야 함
그래서 아래 사진과 같이 다운로드를 하는 과정이 존재



- 본 프로젝트에서는 하나의 기기만 동영상 파일을 내장하고
다른 기기는 해당 파일 없이 재생 가능

Risk Analysis & Risk Reduction Plan

- 영상 재생 동안 기기 사이의 싱크 불일치
- 다른 무엇보다도 신경써야 할 부분으로
네트워크 관련 서적 및 소스코드 참고

Success Criteria

- 영상 재생시 3개 화면의 응답 속도 5ms 미만





KUORA

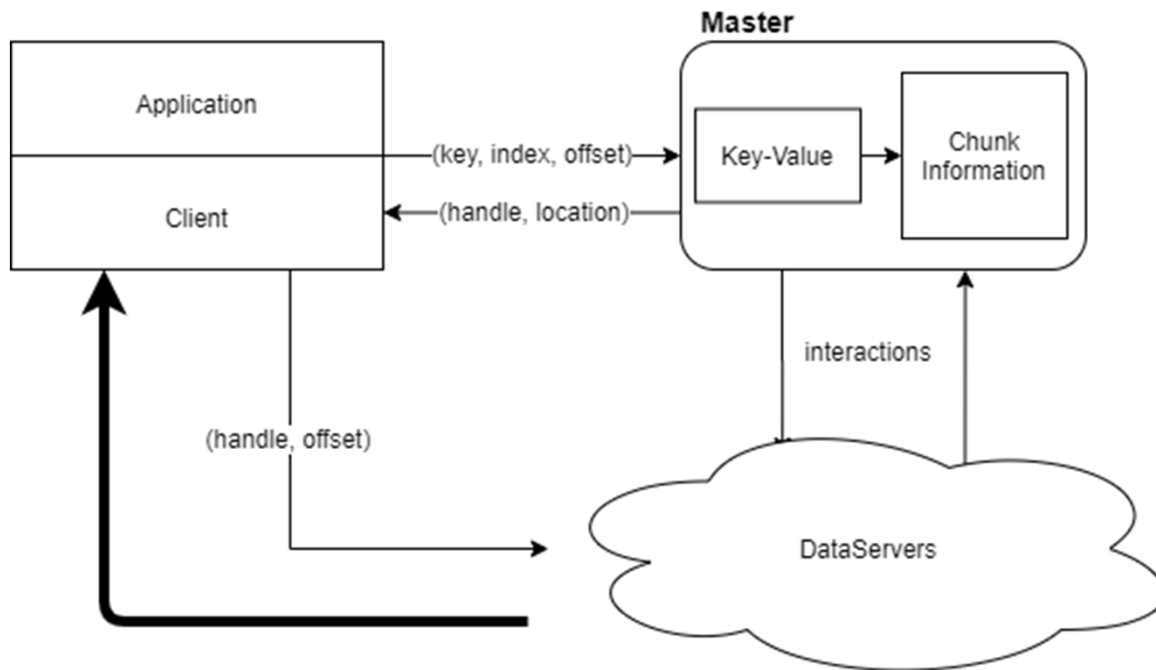
소셜 서비스를 위한 분산 스토리지
(KU + stORAge)

9 팀 (서지원, 홍나리) | 졸업프로젝트 1 | 유준범 교수님

2020.04.04

Concept

KUORA



Assumption

- No modify to files (media)
- Metadata
 - Key-Value
 - Hot data : in-memory
 - Cold data : disk

Alternative Solutions

Another Distributed Storage

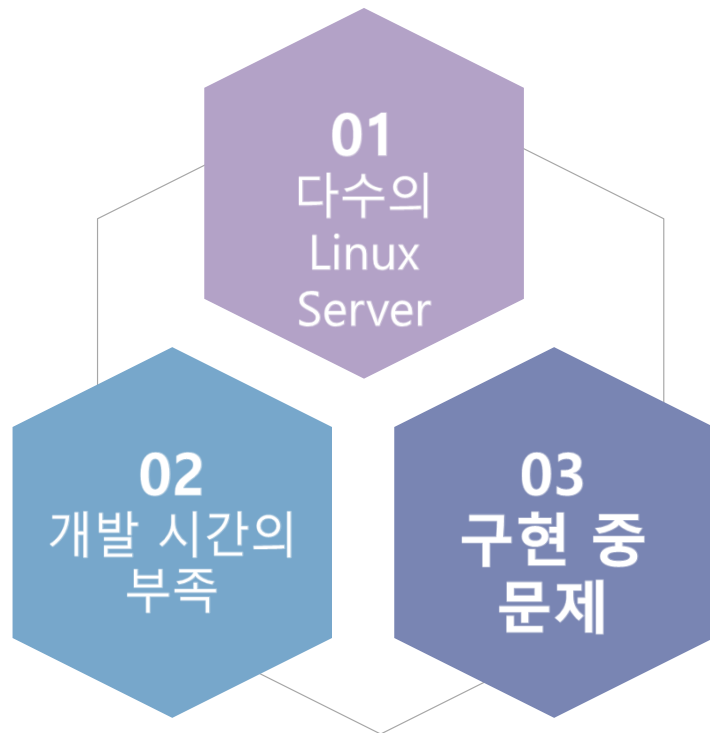


haystack



Risk Analysis

+ Reduction Plan



01

여러 대의 Linux Server가 필요에 따른 비용적 문제.

02

실질적인 개발 기간이 부족할 가능성이 높음. 방대한 내용이므로 선택이 필요. 코로나로 인해 피드백이 어려운 문제도 존재.

03

대표적인 문제로 연산간의 Lock 문제가 있을 수 있음.

Success Criteria

KUORA

With
Go Test Code

1

Basic Operation

Read / Write / Delete 의 작동
Heartbeat 포함 구현 요소의 작동
(Create 시 Replication / Re-Replication
등)

2

Migration

Hot Data의 Expire
및
Cold Storage로의 Migration