



FACE EATING GAME!

TEAM OPEN_THE_DOOR

▶ 1P - 2017112065 김예지

▶ 2P - 2017112082 김혜지

▶ 3P - 2017112106 신소희

▶ 4P - 2017112076 이미란

[INDEX]

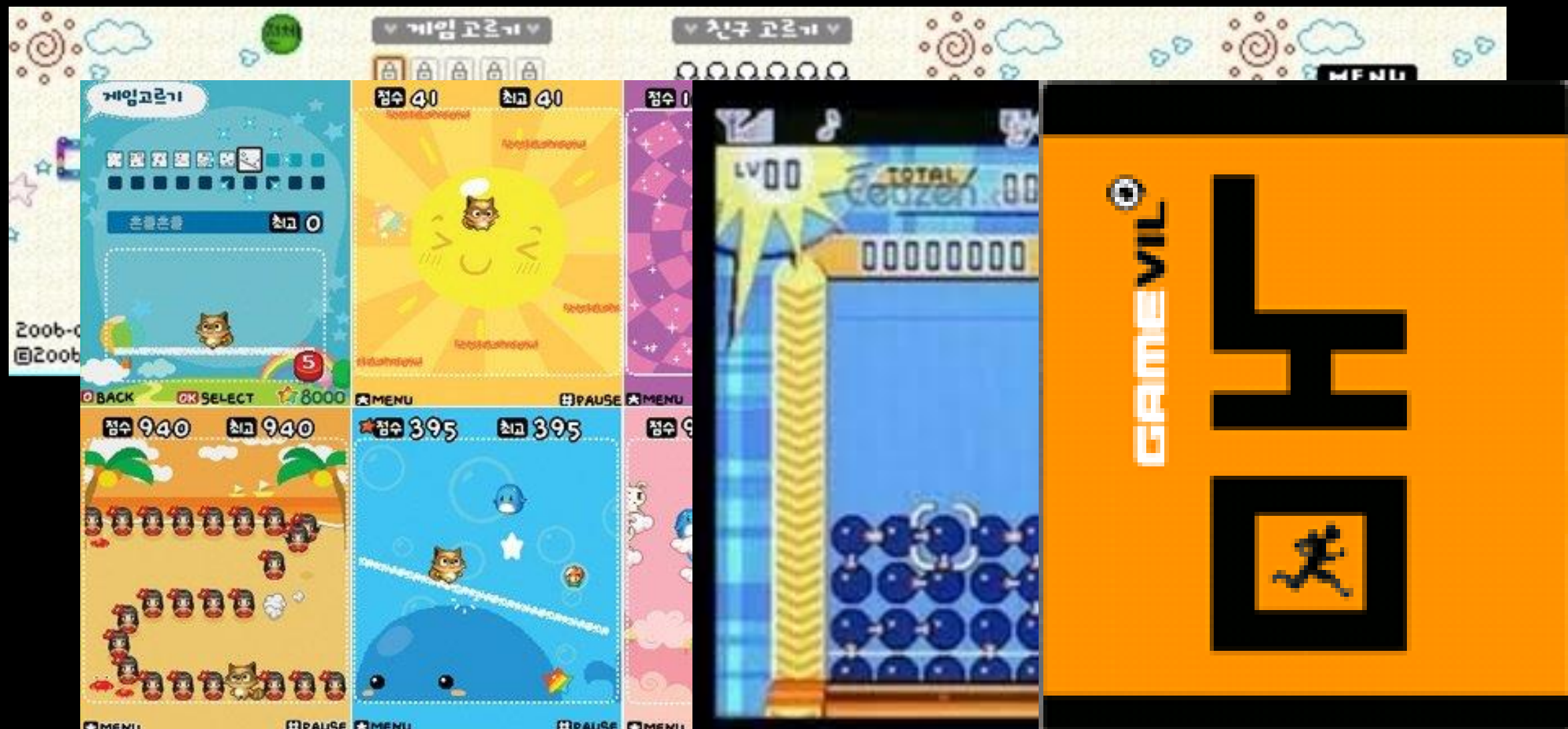


- ▶ STAGE 01 문제 분석
- ▶ STAGE 02 설계 내용
- ▶ STAGE 03 실험 내용
- ▶ STAGE 04 결과물 설명
- ▶ STAGE 05 시연
- ▶ STAGE 06 결론



STAGE 1 문제 분석

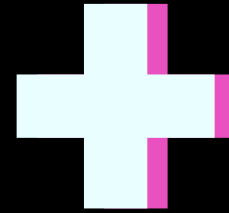
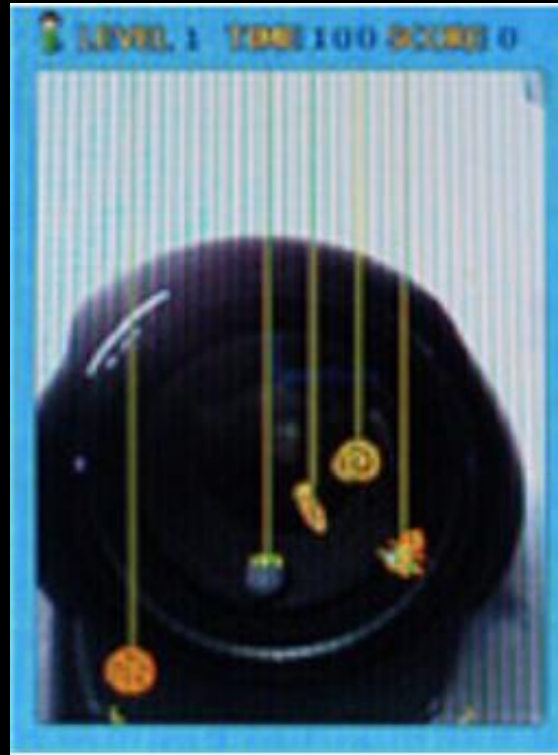
OPENTHEDOOR



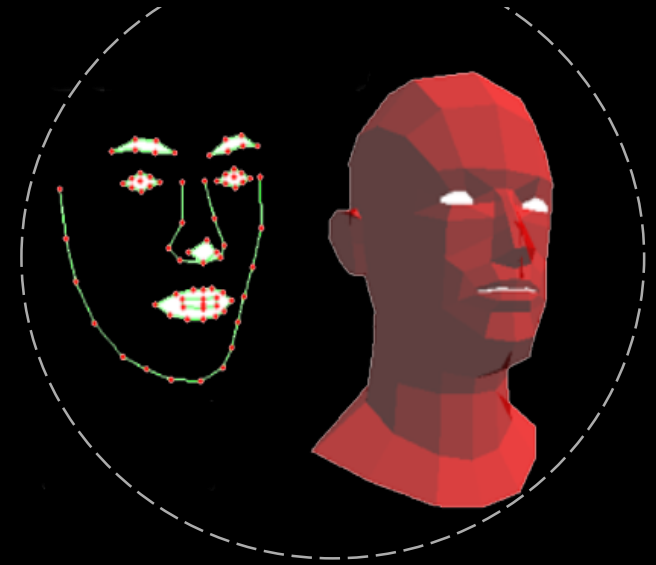


STAGE 1 문제 분석

OPENTHEDOOR 



open source



RETRO



[개선 내용]

無 有



미적 요소

메커니즘

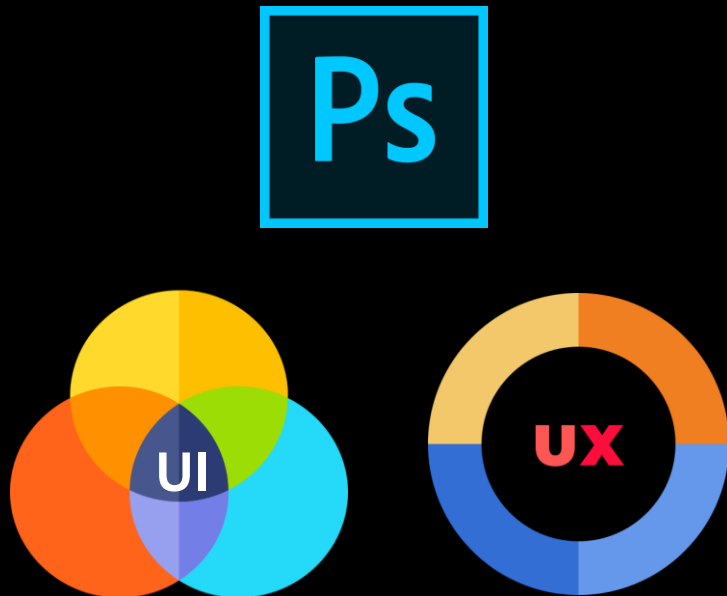
게임의 네 가지 요소

이야기

기술



미적요소



UI & UX

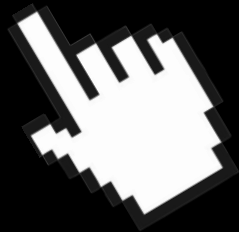


SOUND



미적 요소

메커니즘



게임의 네 가지 요소

이야기

기술



STAGE 2 설계 내용

OPENTHEDOOR



매커니즘 : 절차와 규칙, 목표를 줌

sign up & sign in

game start

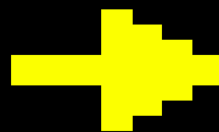
game end

rank



매커니즘

단순한 게임 진행?



흥미가 점차 떨어짐



[Rank System 도입]



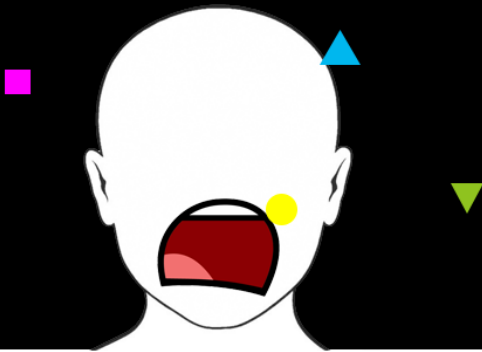


매커니즘


GAME RULES


GAME RULES


1. 날라오는 다양한 도형을 먹으면 점수가 올라갑니다!




GAME RULES

 : 50점 획득

 : 30점 획득

 : 생명 증가

 : 생명 감소

2. 스코어가 높아질수록 속도가 빨라집니다!

GAME RULES



3. 주어진 생명은 3개이고, 최대 3개 이상의 생명을 획득할 수 없습니다. 생명이 0이되면 게임은 종료됩니다.



STAGE 2 설계 내용

OPENTHEDOOR



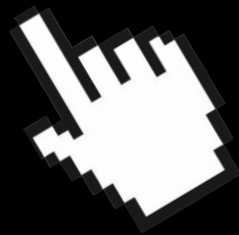
미적 요소

메커니즘

게임의 네 가지 요소

이야기

기술

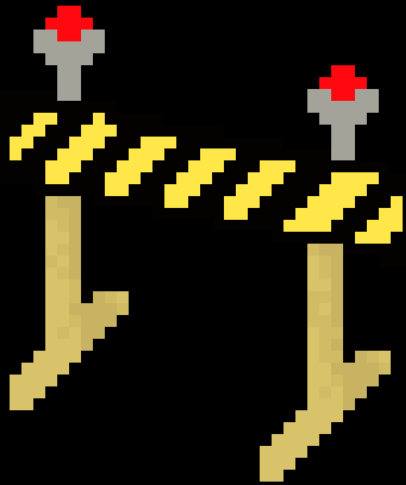




이야기

: 게임에 펼쳐지는 스토리

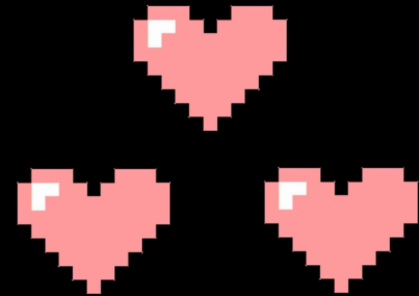
게임이 진행될수록 **긴박**해지도록



[장애물]



[피벗 타임]



[3개의 생명]



미적 요소

메커니즘

게임의 네 가지 요소

이야기

기술





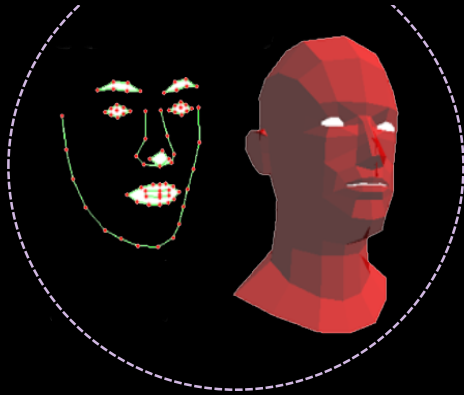
STAGE 2 설계 내용

OPENTHEDOOR



기술

FACE TRACKER



UBUNTU 환경 기반

Open GL



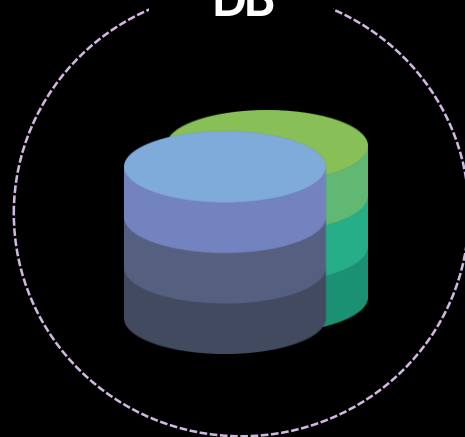
그래픽 라이브러리

Open CV



영상 처리 라이브러리

DB



USER INFO, SCORE 기록

SERVER

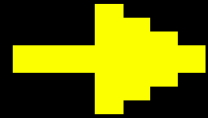
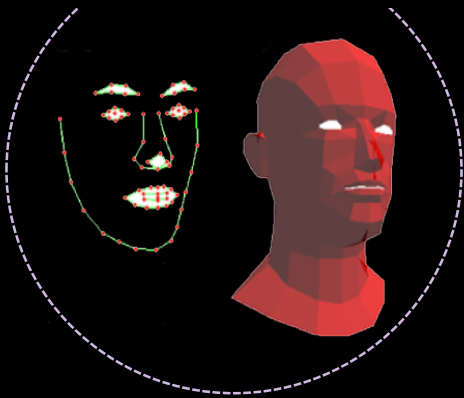


통신 & VIEW



기술

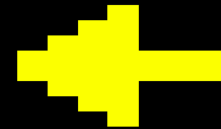
FACE TRACKER



사람의 얼굴을 식별 -> 입을 인식 가능

웹카메라 인식 및 영상 처리

C++ 코드에 라이브러리 추가

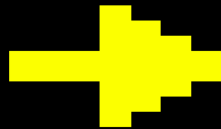


Open CV





기술

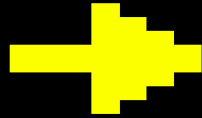
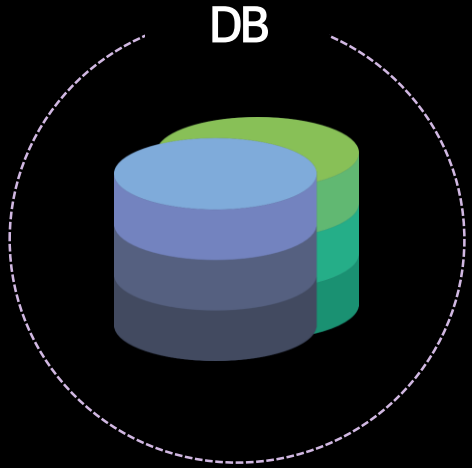


날라오는 도형 및 그래픽 처리

C++ 코드에 라이브러리 추가



기술



User Information, Score

amazons RDS



여러 유저의 정보를 전송하고 전송받음

amazons AWS server, Node js



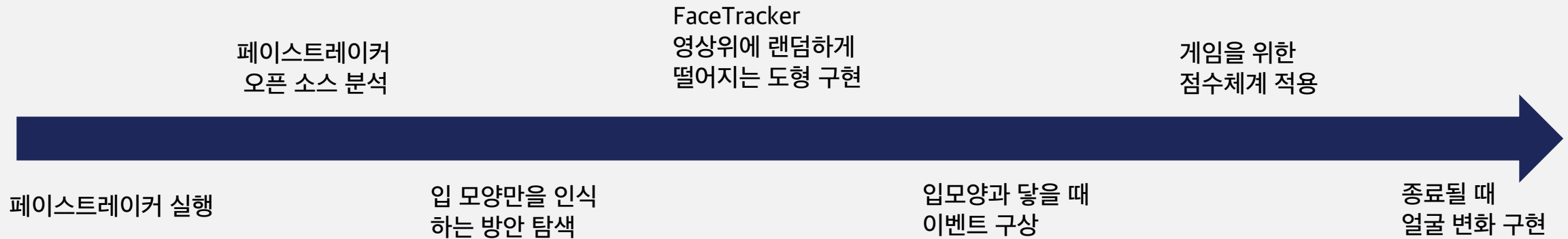


STAGE 2 설계 내용

OPENTHEDOOR



[설계 계획]



[추가 계획]





STAGE 2 설계 내용

OPENTHEDOOR



[설계 계획]



[추가 계획]





Face Tracker

: 디지털 이미지 또는 비디오 프레임에서 사람의 얼굴을 식별하거나 확인하는 수단

분석

▶ 파일 구조

Face Tracker

-> bin : 실행파일

-> include : 헤더파일(.h)

-> src : exe(face_tracker.cc), lib(여러 .cc 파일)

▶ 코드 분석 (face_tracker.cc)

▶ 라이브러리 분석

FACE TRACKER

파일구조

FaceTracker

↳ bin - 실행파일

↳ include - 헤더파일(.h)

↳ src

↳ exe

↳ face_tracker.cc

↳ lib

여러 .cc 파일

face_tracker.cc

: 메인문 포함

• void draw

: 화면에 출력될 선, 점, 곡선
Trangulation을 이용해 거리측정해 위치

• int parse_cmd

: 패이스 트래커 실행시 옵션을 줬다면
해당하는 명령수행

```

$ ./face_tracker --help
face_tracker: Written by Jason Saragih 2010
Performs automatic face tracking

usage: ./face_tracker [options]

Options:
  -h, --help            display this help message
  -m, --model MODEL     Tracker model (default: ../model/face2.tracker)
  -c, --connectivity CONNECTIVITY
                        Connectivity (default: ../model/face.com)
  -t, --triangulation TRIANGULATION
                        Triangulation (default: ../model/face.tri)
  -s, --scaling SCALING
                        Image scaling (default: 1)
  -d, --detections DETECTIONS
                        Frames/detections (default: -1)
  -c, --check           Check for failure
    
```

→ 경로를 따로 지정하는 것인 것 같다. 쓸 필요 없을 듯

• main → 우리는 여기에 코드 추가해야 함!

: 카메라를 실행시키고 화면에 출력

• 헤더파일

Tracker.h : 페이스 트래커와 관련된 모든 함수 (lib에 구현)

opencv/highgui.h

opencv가 제공하는 그래픽 관련 함수

include

공통으로 쓰는 함수 → Init, Load, Save, Write, Read

Fcheck.h

→ 트레이닝 실패 체크

Fdet.h

→ OpenCV의 face detector를 강변
이용해서 FaceTracker가 사용할 수 있게

IO.h

→ Input / Output operation

PAW.h

→ 조각처럼 만들음 → main에서 사용

PDM.h

→ 3D 분산 모델

CLM.h

→ 제공된 Local Model → patch expert

Include 관계

Tracker

→ CLM.h, Fdet.h, Fcheck.h

patch, Fdet, PAW, PDM,

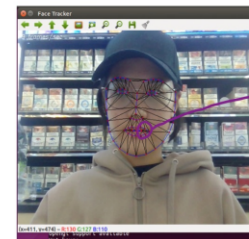
→ IO.h

CLM

→ PDM.h, patch.h

Fcheck

→ PAW.h



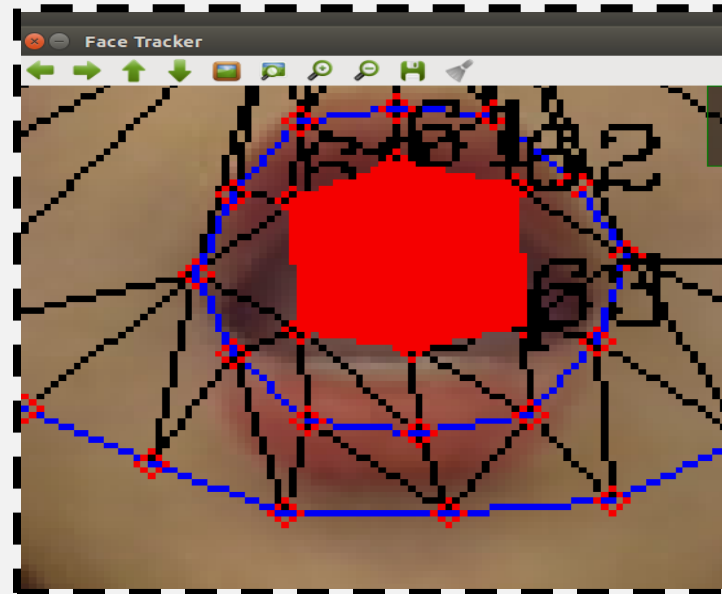
→ main의 draw
함수에서 check로
받아오는 부분, 빨간
점들로 받아와서
크기변화를 측정하
도록 구현해야
할 것 같다.

Face Tracker

입모양인식구체화



얼굴 전체 인식



변화되는 입 모양 인식



Face Tracker

입 영역과 물체가 닿았을 때 이벤트 구현



입 영역의 인덱스 좌표 저장



좌표 영역 안으로 들어올 때 점수 증가



Face Tracker

페이스 가이드라인 설치



게임의 영역을 제한

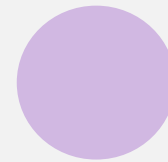


밖으로 나가게 되면 경고를 표시
페이스 트래커 재인식



Face Tracker

랜덤하게 날아오는 도형 구현



랜덤한 위치에서, 랜덤한 간격, 랜덤한 색상으로 날라오는 도형



SERVER



[팀원 A]

what is nodejs ...?



AWS RDS





SERVER

Sign up & Sign in 시 비밀번호 암호화 (DB)

- **crypto 모듈** : 문자열을 암호화, 복호화, 해싱하는 Nodejs 기본 내장 모듈
- Hashing Algorithm으로 **SHA512**를 사용
- 보안을 위해 Salting + Key Stretching 함께 사용



```
const salt = await crypto.randomBytes(32);
const hashedpwd = await crypto.pbkdf2(user_pw, salt.toString('base64'), 100000, 'sha512');
console.log('hashing success');
// password 해싱
let insertUserQuery = `INSERT INTO openthedoor.user (user_id, user_pw, user_salt, user_idx, score) VALUES ('${user_id}', '${hashedpwd.toString('base64')}', '${salt.toString('base64')}', 0, 0);`;
let insertUserResult = await db.queryParam_Arr(insertUserQuery, [user_id, hashedpwd.toString('base64'), salt.toString('base64')]);
```

Result Grid						Filter Rows:	Edit:	Export/Import:	Wrap Cell Content:
user_id	user_pw	user_salt	user_idx	score					
yeji	OTt9Ur7vdOvMDLu4GleBTHagWKiIFTHR1ZOMN...	IPjm482eKobrVxqKarc+ZUYIGIsW15FNCtxIcRA84aY=	40	40					
ossp7	8cr8IAo/YK1E3067G01Bcv8zu3xGYRdE2ZJriaITs...	SGeh5cxbr1yQEFFF65TjBUOaCdrsdJ8pxxyDOyE98=	41	40					
miranlee	oyhnPIgrNrnUk/ujBZVZTa+W5SKomnz210RHX...	IFB5gCd0gOXUTwunVG23bSB13ISkEQEU+aPS24gX/dw=	43	310					
hyex	WCcAR049vrJzBj4dUVk+HYg6B3tVwA/eQP0PH...	AdzcwYaQscX8E8dM3S8hQOQE0R4wvvyZF66q2vfmMLI=	44	390					
miran	qOLY8HWeUFZ3tv/1hl+Qhf9RBb5f8A1WHZVDJ...	M5BnUgL1Su1kUPwJruEknhlKMgJU+G3iINLAQUyL8=	50	40					
sun	1ny6CQOfvZiVJAOnVGrARRCyzwBAmUP0udHs...	X59Xp/pvBfW6rxj8JVoH9I7xMjUgneT2fa05kymo4vo=	52	40					
sonbug	tmltzjghCp0y8ChI9P0ufJ6YDrBafGI1iWaByiF3A...	+zskMMttCr867Vv8Yd++XxYIjWDCSMT54AWx7KwFnY=	55	200					



SERVER

Sign up & Sign in 시 비밀번호 암호화 (view)

- 사용자가 보는 시야에서도 비밀번호를 **** 처리
- 사용자를 제외한 타인이 비밀번호를 알 수 없어 보안성을 높임

-----LOGIN-----

ID : sonbug

PW : ***

[1]signup [2]signin : 1



SERVER

예외 처리

- 존재하는 회원의 아이디로 회원가입 요청 시
- 존재하지 않은 아이디로 로그인 시도 시
- 틀린 비밀번호 입력 시

-----LOGIN-----

ID : sonbug
PW : ****

[1]signup [2]signin : 1

already exist user!

-----LOGIN-----

ID : sonbug
PW : ****

[1]signup [2]signin : 2

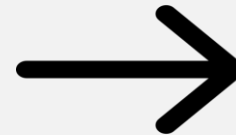
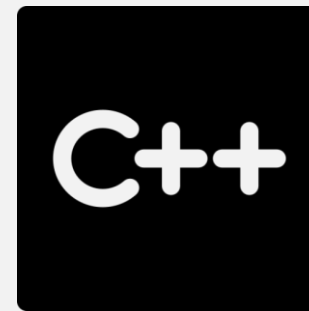
pwd error !



SERVER

SOCKET PROGRAMMING

- node.js 와 C++코드를 연결해줄 매개체가 필요
- SOCKET 을 이용하여 해결



Socket





STAGE 3 실험 내용

OPENTHEDOOR



현동

2019-1-OSSP2-Openthedoor-7

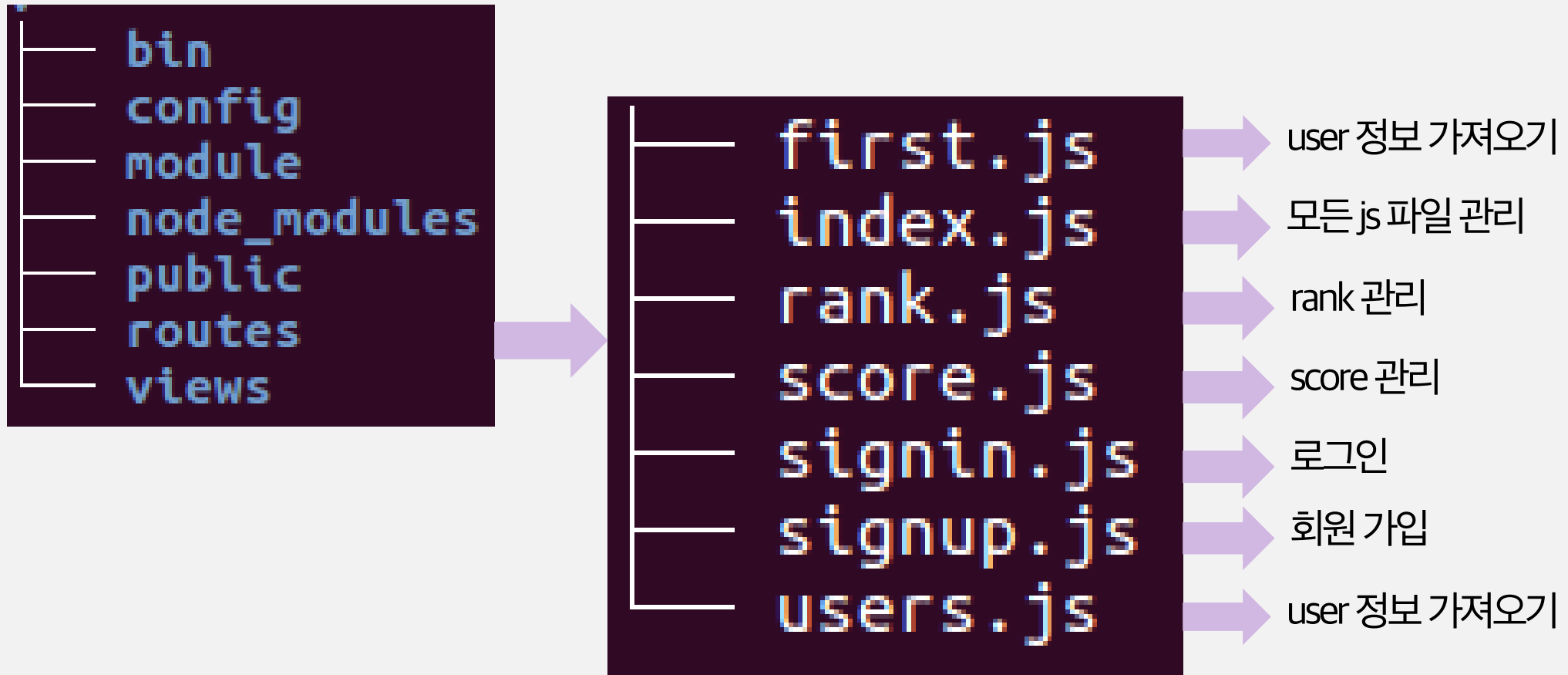
"YAM-YAM" Game program using Face Tracker



<div><div></div><div>C++</div></div> <div><div></div><div>0</div></div> <div><div></div><div>0</div></div> <div><div></div><div>0</div></div> <div><div></div><div></div></div> <div><div></div><div>0 Open</div></div> <div><div></div><div>7 Closed</div></div>	<div>Author ▾</div> <div>Labels ▾</div> <div>Projects ▾</div> <div>Milestones ▾</div> <div>Assignee ▾</div> <div>Sort ▾</div>
<div><div></div><div><div></div>회원가입, 로그인에 관한 예외처리 필요</div><div>#8 by shinsohui was closed 3 days ago</div></div>	<div><div></div>1</div>
<div><div></div><div><div></div>Cannot read property 'toString' of undefined</div><div>#7 by hyex was closed 13 days ago</div></div>	<div><div></div>1</div>
<div><div></div><div><div></div>random increases in scores when shapes are in contact with the mouth. <u>wontfix</u></div><div>#6 by rannan09 was closed 3 days ago</div></div>	<div><div></div>1</div>
<div><div></div><div><div></div>Print life symbol to display</div><div>#5 by rannan09 was closed 3 days ago</div></div>	<div><div></div>2</div>
<div><div></div><div><div></div>linking c++ & node.js</div><div>#4 by hyex was closed 16 days ago</div></div>	<div><div></div>1</div>
<div><div></div><div><div></div>coordinate system unification</div><div>#3 by rannan09 was closed 20 days ago</div></div>	<div><div></div>2</div>
<div><div></div><div><div></div>Coordinates Problem</div><div>#1 by rannan09 was closed 20 days ago</div></div>	<div><div></div>1</div>

진행결과

Openthedoor_Server의 디렉토리 구조





STAGE 4 결과물 설명

OPENTHEDOOR



SIGN UP/IN

INTRO

GAME RULES

GAME

END

RANK

```
./face_tracker /home/mi  
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)  
  
-----LOGIN-----  
  
ID : miranlee  
PW : ****  
  
[1]signup [2]signin :
```



STAGE 4 결과물 설명

OPENTHEDOOR



SIGN UP/IN

INTRO

GAME RULES

GAME

END

RANK

YAH-YAH
EATING GAME

START

game rules

RANKING

EXIT

MADE BY OPEN_THE_DOOR



STAGE 4 결과물 설명

OPENTHEDOOR



SIGN UP/IN

INTRO

GAME RULES

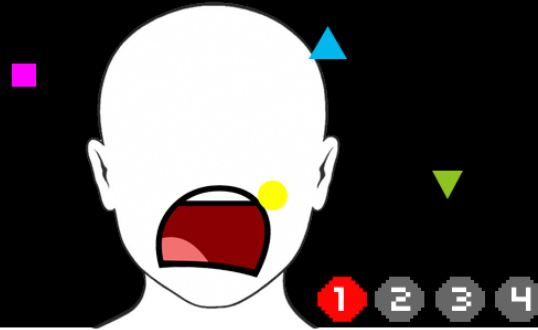
GAME

END

RANK


GAME RULES


1. 날라오는 다양한 도형을 먹으면 점수가 올라갑니다!





GAME RULES

2. 스코어가 높아질수록 속도가 빨라집니다!

 : 50점 획득

 : 30점 획득

 : 생명 증가

 : 생명 감소



GAME RULES



3. 주어진 생명은 3개이고, 최대 3개 이상의 생명을 획득할 수 없습니다. 생명이 0이되면 게임은 종료됩니다.



GAME RULES



4. 게임이 종료되면
최고 점수가 기록됩니다.
친구와 점수를 겨뤄보세요!





STAGE 4 결과물 설명

OPENTHEDOOR



SIGN UP/IN

INTRO

GAME RULES

GAME

END

RANK





STAGE 4 결과물 설명

SIGN UP/IN

INTRO

GAME RULES

GAME

END

RANK

OPENTHEDOOR





STAGE 4 결과물 설명

OPENTHEDOOR



SIGN UP/IN

INTRO

GAME RULES

GAME

END

RANK

GAME RANK					
RANKING					
miranlee 님의 최고점수: 310					
▶ 1-10위 ▶ 10-20위 ▶ 20-30위					
1	hyex	390	6	miran	40
2	miranlee	310	7	sun	40
3	sonbug	200	8	sjn	40
4	yeji	40	9	ran	0
5	ossp7	40	10	rann	0



STAGE 5 시연

OPENTHEDOOR



시연



소감

Face Tracker

이번 공개SW 프로젝트를 통해 OpenGL과 OpenCV에 관한 깊이 있는 지식을 배웠고, 두 개를 이용한 어떤 프로그램이든 만들 수 있을 것같은 자신감이 생겼다. 게임을 제작해보면서 게임에 관한 다양한 설계를 통해 팀원 간에 여러 아이디어를 생각해보며 재밌게 프로젝트를 진행하였다.



소감

Server

웹 프로그래밍때 로컬 호스트 서버를 다뤄본 것을 제외하고 서버를 다뤄본 적이 없었는데 이번 프로젝트를 계기로 서버에 대해 전반적으로 배우게 된 시간이었다. 짧은 시간 동안 언어가 어떤 구조를 가지며 어떤 형식으로 쓰이는 지에 대해 파악하기가 힘들었지만 앞으로 다른 프로젝트를 도움이 많이 될 것 같다.



소감

OSSP

오픈소스를 처음 사용해봤는데 흥미로운 오픈 소스가 많아서 주제 고르기가 힘들었다. 이 오픈 소스처럼 큰 프로젝트를 하나를 참고해 프로젝트를 진행한 경험 많지 않았는데, 기존의 오픈 소스를 분석하며 정말 아무 것도 없는 오픈 소스에서 이것 저것 기능을 추가하며 프로그램이 발전되는 것을 보니 뿌듯하였다.



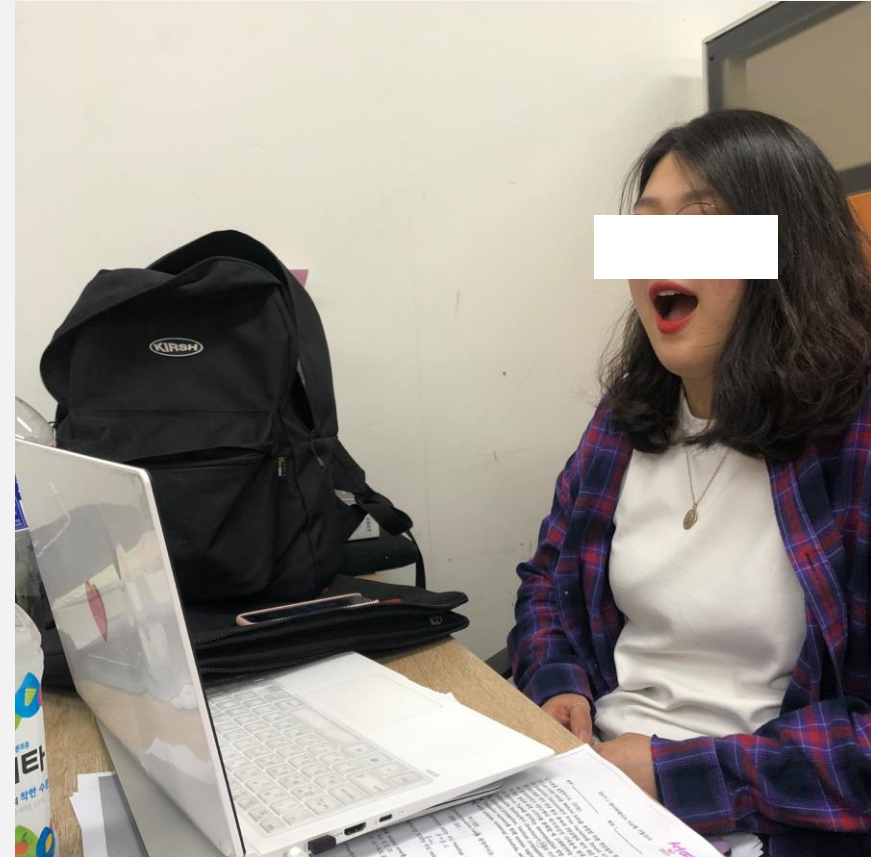
STAGE 6 결과

OPENTHEDOOR



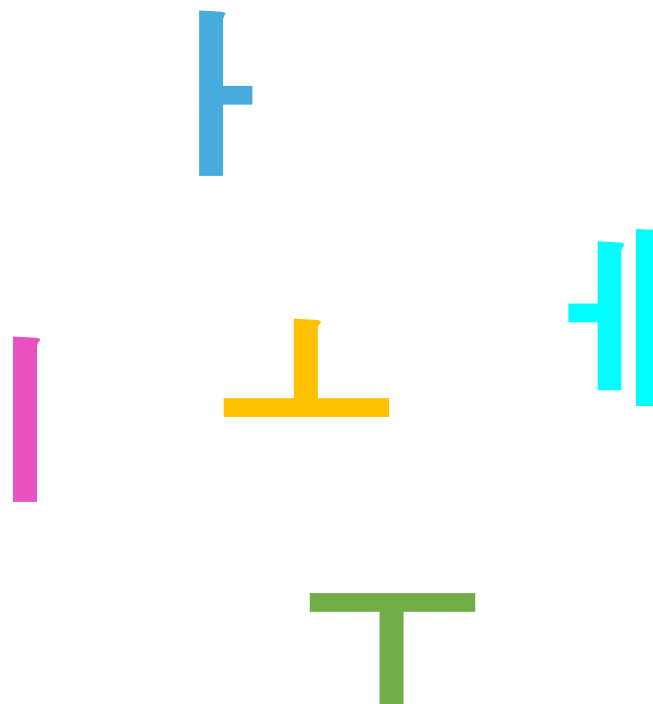
소감

어릴 적 운동회 때 하던, 과자 먹기 게임이 생각나고 얼굴을 인식해서 할 수 있어서 너무 흥미로웠다. 그리고 게임을 여러 번하고 싶도록 하는 요소들이 많아서 계속 하고 싶게끔 하는 재미있는 게임이었다. 이 게임이 실제로 출시된다면, 중독돼서 이번 생은 망할 거 같다 ^_^ 그 정도로 재밌었다 ! 정 ~ 말 ~ 잘 만들었다 !





향후 계획 청각 장애인의 발화를 도와주는 게임으로 발전



Q & A

YAM-YAM GAME

TEAM OPEN_THE_DOOR