



국민대학교  
소프트웨어융합대학  
소프트웨어학부


# 캡스톤 디자인 I

## 종합설계 프로젝트

|        |         |
|--------|---------|
| 프로젝트 명 | ON감독    |
| 팀 명    | 캡스톤 19팀 |
| 문서 제목  | 중간보고서   |

|         |            |
|---------|------------|
| Version | 1.0        |
| Date    | 2022-04-04 |

|      |           |
|------|-----------|
| 팀원   | 한 창희 (조장) |
|      | 장 우석      |
|      | 최 근표      |
|      |           |
|      |           |
| 지도교수 | 이 창우 교수   |

|  |                         |             |            |
|--|-------------------------|-------------|------------|
|  <b>국민대학교</b><br><b>소프트웨어학부</b><br><b>캡스톤 디자인 I</b> | <b>중간보고서</b>            |             |            |
|  | <b>프로젝트 명</b>           | ON감독        |            |
|  | <b>팀 명</b>              | 캡스톤 19팀     |            |
|  | Confidential Restricted | Version 1.0 | 2022-04-04 |

### CONFIDENTIALITY/SECURITY WARNING

이 문서에 포함되어 있는 정보는 국민대학교 소프트웨어융합대학 소프트웨어학부 및 소프트웨어학부 개설 교과목 캡스톤 디자인 수강 학생 중 프로젝트 "ON감독"을 수행하는 팀 "캡스톤 팀19"의 팀원들의 자산입니다. 국민대학교 소프트웨어학부 및 팀 "캡스톤 19팀"의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

## 문서 정보 / 수정 내역

|                 |                |
|-----------------|----------------|
| <b>Filename</b> | 중간보고서-ON감독.doc |
| <b>원안작성자</b>    | 한창희, 장우석, 최근표  |
| <b>수정작성자</b>    | 한창희, 장우석, 최근표  |

| 수정날짜       | 대표수정자 | Revision | 추가/수정 항목 | 내 용                               |
|------------|-------|----------|----------|-----------------------------------|
| 2022-03-30 | 장우석   | 1.0      | 최초 작성    | 프로젝트 목표, 수행 내용 및 중간결과, 향후 추진계획 작성 |
| 2022-04-02 | 장우석   |          | 내용 추가    | 2.2.3 , 2.2.4                     |
| 2022-04-03 | 최근표   |          | 내용 추가    | 2.2.5                             |
|            | 한창희   |          | 내용 수정    | 수행 내용 및 중간결과                      |
|            | 최근표   |          | 내용 추가    | 4.1.1 ~ 4.1.4                     |
|            | 한창희   |          | 최초 작성    | 수정된 연구내용 및 추진 방향                  |
|            |       |          |          |                                   |
|            |       |          |          |                                   |
|            |       |          |          |                                   |

|  |                         |             |            |
|--|-------------------------|-------------|------------|
|  <b>국민대학교</b><br><b>소프트웨어학부</b><br><b>캡스톤 디자인 I</b> | <b>중간보고서</b>            |             |            |
|  | <b>프로젝트 명</b>           | ON감독        |            |
|  | <b>팀 명</b>              | 캡스톤 19팀     |            |
|  | Confidential Restricted | Version 1.0 | 2022-04-04 |


## 목 차

|       |                         |    |
|-------|-------------------------|----|
| 1     | 프로젝트 목표 .....           | 4  |
| 2     | 수행 내용 및 중간결과 .....      | 5  |
| 2.1   | 계획서 상의 연구내용 .....       | 5  |
| 2.2   | 수행내용 .....              | 5  |
| 2.2.1 | 서비스가 제공하는 기능 .....      | 5  |
| 2.2.2 | 해당 기술을 사용한 이유 .....     | 6  |
| 2.2.3 | Project Flow .....      | 9  |
| 2.2.4 | Eye Tracking Flow ..... | 10 |
| 2.2.5 | 중간보고서 결과물 .....         | 11 |
| 3     | 수정된 연구내용 및 추진 방향 .....  | 12 |
| 3.1   | 수정사항 .....              | 12 |
| 3.1.1 | 22/03/11 .....          | 12 |
| 3.1.2 | 22/03/25 .....          | 13 |
| 3.1.3 | 22/04/01 .....          | 13 |
| 4     | 향후 추진계획 .....           | 14 |
| 4.1   | 향후 계획의 세부 내용 .....      | 14 |
| 4.1.1 | 로그인 기능 .....            | 14 |
| 4.1.2 | 신분증 대조 기능 .....         | 14 |
| 4.1.3 | 영상 저장 기능 .....          | 14 |
| 4.1.4 | IP 위치 파악 기능 .....       | 14 |
| 5     | 고충 및 건의사항 .....         | 15 |

|  |                         |             |            |
|--|-------------------------|-------------|------------|
|  <b>국민대학교</b><br><b>소프트웨어학부</b><br><b>캡스톤 디자인 I</b> | <b>중간보고서</b>            |             |            |
|  | <b>프로젝트 명</b>           | ON감독        |            |
|  | <b>팀 명</b>              | 캡스톤 19팀     |            |
|  | Confidential Restricted | Version 1.0 | 2022-04-04 |

## 1 프로젝트 목표

코로나바이러스 감염증(COVID-19)으로 인하여 비대면 활동이 활성화되었다. 대학에서는 다양한 평가시험이 진행되는 가운데, 집단의 부정행위를 파악해내기 힘들어 어려움을 겪고 있다. 이에 따른 다양한 해결방안 중 하나는 절대평가를 예로 들 수 있다. 시험을 치르는 학생의 점수를 집단이 정해놓은 절대적인 기준에 따라 평가를 하는 것이다. 따라서 부정행위가 일어났을 시 받는 학생들의 피해를 상쇄하여 피해를 줄이고 있다. 하지만 이 방법은 근본적인 해결방안이 될 수 없다. 시험의 공정성에 문제가 생기게 되고 전체적인 학생들의 학업 분위기에 영향을 미칠 수 있다. 따라서 비대면 시험에서 일어날 수 있는 부정행위를 적발하고 그에 맞는 제재를 가할 수 있도록 운영해야 한다. 이에 ON 감독 서비스는 Web Real-Time Communication 인 WebRTC 를 이용하여 웹에서 비대면 시험을 치를 수 있는 환경을 제공하고, MediaPipe facemesh model 을 사용해 Eye Tracking algorithm 을 적용하여 부정행위를 적발한다.

|  |                         |             |            |
|--|-------------------------|-------------|------------|
|  <b>국민대학교</b><br><b>소프트웨어학부</b><br><b>캡스톤 디자인 I</b> | <b>중간보고서</b>            |             |            |
|  | <b>프로젝트 명</b>           | ON감독        |            |
|  | <b>팀 명</b>              | 캡스톤 19팀     |            |
|  | Confidential Restricted | Version 1.0 | 2022-04-04 |

## 2 수행 내용 및 중간결과

### 2.1 계획서 상의 연구내용

ON 감독 서비스가 학생들에게 제공하는 것은 다양성, 공정성, 포용성이다. 상대평가를 진행할 수 없어서 서술식 답안이나 절대평가라는 획일화된 시험 방식으로 나아갈 수 있는 환경에서 조금이라도 다양한 시험방식을 통해 교수(관리자)는 다양한 성과지표를 받을 수 있다. 또한 Eye Tracking 이나 얼굴각도, 사용자의 위치를 통해 부정행위를 파악하여 불확실한 환경에서 공정성을 높이고 있다. 마지막으로 직접 대면 시험을 보기 어려운 상황에 놓인 학생들에게 시험을 치를 수 있는 환경을 조성하여 조직 구성원들을 포용하는 포용성을 높일 수 있다.

개발자 교육, 평가, 채용 플랫폼인 프로그래머스의 경우 이러한 부정행위를 관리해주는 프로그램이 존재한다. 이것이 바로 대표적인 온라인 시험 감독 solution 인 ‘모니토’이다. 모니토의 경우에도 온라인 화상 시험 서비스에서 공정성과 전문성을 높이고 있고 화면공유 및 비디오 녹화를 통해 이를 해결하고 있다.


하지만 이러한 기존 제품들의 경우 한계가 존재한다. 다양한 각도의 카메라가 존재하지만 사용자의 움직임 등을 통해 부정행위를 파악하기 어렵고, 응시자가 많으면 많을수록 이를 파악하는 관리자의 수가 많아져 하기 때문에 인원, 비용 등의 번거로움을 겪을 수 있다. 따라서 ON 감독 서비스는 MediaPipe 에서 제공하는 FaceMesh 를 통해 얼굴각도, 홍채를 인식하고 추적하여 부정행위를 잡아내고, 이를 관리자에게 알람이나 화면의 감박임을 통해 전송하여 관리자의 인원 문제에도 효과적으로 대응할 수 있게끔 한다.

### 2.2 수행내용

#### 2.2.1 서비스가 제공하는 기능

먼저 부정행위를 감지하는 서비스 제공에 필요한 기술들이 무엇인지 찾아보았다. 팀19는 크게 다음과 같이 정리한다.

1. 감독관과 시험 응시자의 채팅과 같은 소통 기능
2. 실시간 오디오 및 영상 전송 기능
3. 신분증을 통한 본인 확인 기능
4. Eye Tracking 을 이용한 치팅 방지 기능

|  |                         |             |            |
|--|-------------------------|-------------|------------|
|  <b>국민대학교</b><br><b>소프트웨어학부</b><br><b>캡스톤 디자인 I</b> | <b>중간보고서</b>            |             |            |
|  | <b>프로젝트 명</b>           | ON감독        |            |
|  | <b>팀 명</b>              | 캡스톤 19팀     |            |
|  | Confidential Restricted | Version 1.0 | 2022-04-04 |

먼저 1, 2번의 경우 WebRTC 를 이용하여 구현할 수 있다. 이는 웹을 통해 사용자의 비디오 및 오디오를 받을 수 있으며 채팅 또한 이용이 가능하다.


4번의 경우 MediaPipe 에서 제공하는 faceMesh 모델을 사용한다. 이를 사용하면 얼굴의 형태를 그물망 형태로 나타내 주고 468개의 얼굴점으로 분할하기 때문에 특정 위치(눈이나 입의 좌표)를 파악하기가 가능하다. 하지만 ON 감독에서 필요로 하는 눈동자 움직임에 따라 부정행위를 감지하는 기능은 구현되어 있지 않기 때문에 직접 구현한다.

## 2.2.2 해당 기술을 사용한 이유

### -WebRTC

먼저 WebRTC 를 설명하기 전 HTTP 에 대한 이해가 선행되어야 한다. 브라우저와 서버는 HTTP 를 이용해 소통할 수 있다. 이는 인터넷 데이터 교환에 있어서 필수적이며, 업로드, 다운로드, 이미지, 문서 등 인터넷에서 오가는 데이터는 http 프로토콜이 필요하다. 클라이언트(브라우저)는 웹 페이지를 열기 위해 웹서버에게 페이지 정보를 요구하는 메시지를 보내며 이것이 http request 요청이다. 해당 서버는 request 를 보고 클라이언트가 페이지 정보를 요구하는 것을 확인한 뒤 http response 를 보내게 된다. 하지만 서버가 클라이언트의 요청에 응답하고 나면 통신이 종료되는데 한번 응답이 이루어지면 서버는 클라이언트에게 무엇도 보낼 수 없다. 따라서 서버가 클라이언트에게 데이터를 보낼 수 있는 건 클라이언트가 정보를 요청한 그 순간이다.

단방향 통신에 대한 해결책으로 WebSocket 이 등장했다. WebSocket 프로토콜은 Request-Response 형태가 아닌 connection 의 open - close 형태로 구분한다. 즉 클라이언트가 WebSocket 을 이용해 서버와 연결을 하면 클라이언트와 서버 간의 통신은 open 상태일 것이다. open 상태가 끊어지기 전까지 해당 통신은 유지되며 close 상태로 변경되었을 시 종료된다. 또한 양방향 통신이기 때문에 통신이 열려있는 상태에서는 계속해서 메시지를 보내는 것이 가능하고 서버도 클라이언트에게 메시지를 자유롭게 보낼 수 있다. 다시 말해 서버는 클라이언트로부터

|  |                         |             |            |
|--|-------------------------|-------------|------------|
|  <b>국민대학교</b><br><b>소프트웨어학부</b><br><b>캡스톤 디자인 I</b> | <b>중간보고서</b>            |             |            |
|  | <b>프로젝트 명</b>           | ON감독        |            |
|  | <b>팀 명</b>              | 캡스톤 19팀     |            |
|  | Confidential Restricted | Version 1.0 | 2022-04-04 |

Request 를 기다릴 필요가 없다. 이 때문에 채팅방이나 주식 현황 같은 정보를 실시간으로 사용하기 위해 WebSoket 을 사용하고 있다.

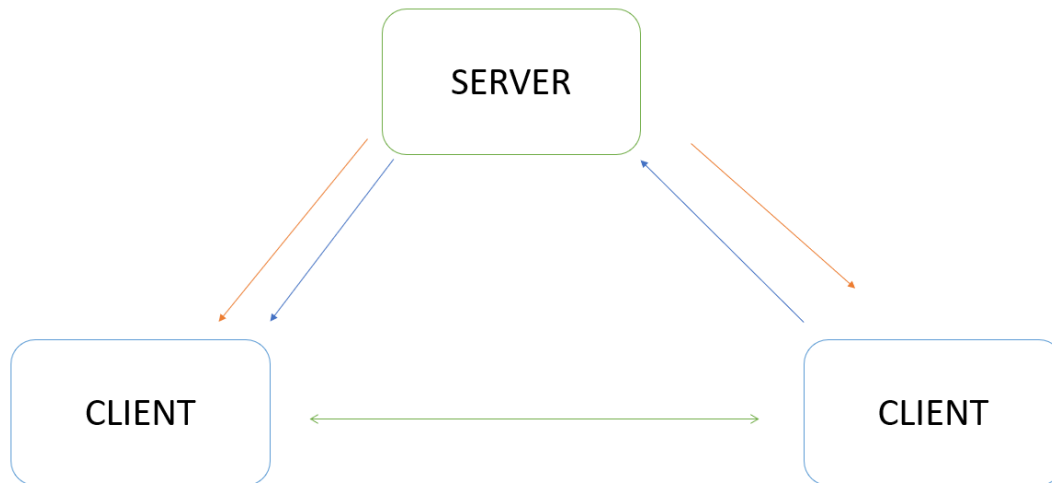
하지만 WebSocket 도 단점이 존재하는데 서버의 메모리 파워가 중요하다는 점이다. 모든 클라이언트는 웹소켓에 연결되고 클라이언트의 메시지가 웹소켓 서버에 전달되면서 다시 서버가 해당 메시지를 클라이언트에 전달한다. 따라서 모든 통신을 추적하기 위해 유저가 많으면 많을수록 더 많은 메모리가 필요하고 서버 비용도 증가한다. 또한 메시지를 받으면 다른 사람들에게 포워딩 해야 하기 때문에 서버도 빠르게 유지해야 한다. 하지만 서버에 connection 이 많다면 딜레이가 발생할 수 있기 때문에 속도가 느려지고 서버가 내려가면 다운되어버리는 문제가 생긴다. 이때부터 클라이언트를 서버에 연결하지 않고 바로 클라이언트와 클라이언트를 연결하기 위해 WebRTC 가 등장했다.

WebRTC 의 장점은 클라이언트의 메시지가 서버를 거치지 않고 바로 클라이언트로 전달되기 때문에 서버를 사용하는 것보다 속도가 빠르다. 또한 단순한 텍스트뿐만이 아니라 영상과 오디오, 채팅 모두 실시간으로 전송할 수 있기 때문에 ON 감독 서비스에 필요한 기술이다.

#### -MediaPipe FaceMesh

먼저 Eye Tracking 을 적용하기 위해서는 얼굴각도, 홍채를 인식하고 추적할 수 있어야 했다. 따라서 ON 감독 서비스에서는 얼굴의 형태를 그물망 형태로 나타내 주고 468개의 얼굴점으로 분할하여 나타내주는 MediaPipe FaceMesh 모델을 적용한다.

### -Node.js




앞서 WebSocket 과는 다르게 서버를 거치지 않고 바로 클라이언트로 전달되는 것이 WebRTC의 장점이라고 언급했다. 이는 클라이언트와 클라이언트의 직접적인 연결이 이루어진 후에 일어난다. 따라서 첫 연결 시 signaling(WebRTC가 연결이 이루어지려면 signaling이라는 작업이 필요하다)이 필요하고, signaling 과정(Communication 세션의 설정 및 컨트롤, 종료되는 과정을 의미)을 위한 서버가 필요한데 ON 감독 서비스에서는 Node.js를 사용한다. Node.js는 실시간 처리 작업을 사용하기에 유용하다. 현재 이 프로젝트에서는 유저들이 메시지를 전송하고 받는 실시간 처리 작업이 존재하기 때문에 Node.js는 좋은 성능을 보일 수 있다.

### -peer.js

WebSocket은 양방향통신을 위한 프로토콜이며 socket.io는 양방향통신을 하기 위해 웹소켓 기술을 활용하는 라이브러리이다. 이와 비슷하게 WebRTC 자체적으로 사용하기에는 개발 과정이 다소 복잡할 수 있는데 WebRTC를 좀 더 쉽게 사용할 수 있게 해주는 라이브러리이다.



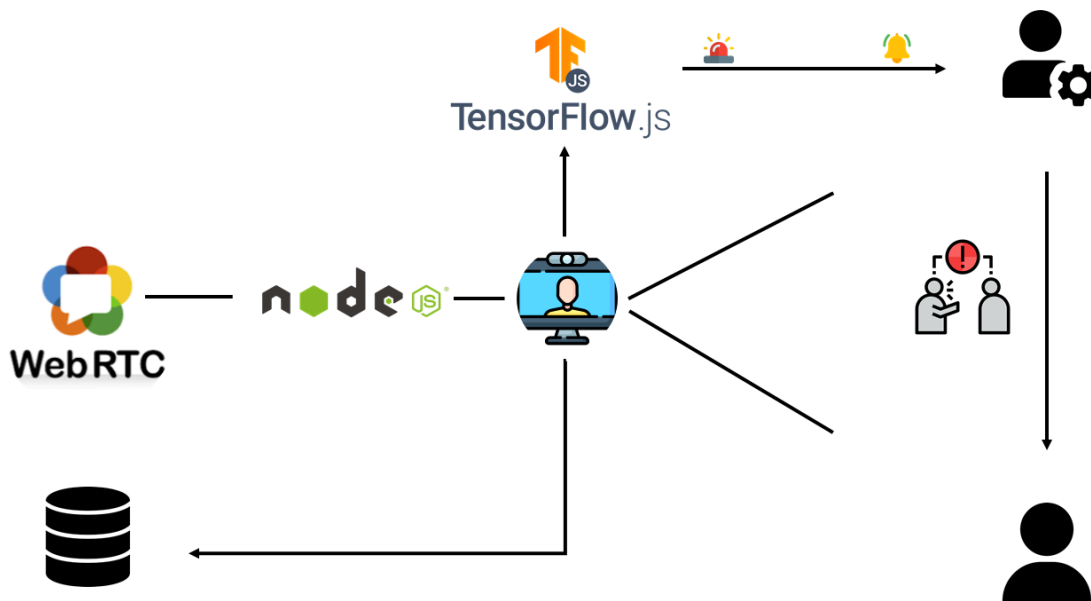
|  |                         |             |            |
|--|-------------------------|-------------|------------|
|  <b>국민대학교</b><br><b>소프트웨어학부</b><br><b>캡스톤 디자인 I</b> | <b>중간보고서</b>            |             |            |
|  | <b>프로젝트 명</b>           | ON감독        |            |
|  | <b>팀 명</b>              | 캡스톤 19팀     |            |
|  | Confidential Restricted | Version 1.0 | 2022-04-04 |

-react

react 는 페이스북에서 만든 웹 프레임워크로 사용자 인터페이스를 만들기 위해 사용한다. react 는 SPA(Single page application)란 개념이 존재하는데 페이지가 하나인 애플리케이션을 말한다.


예전에는 A.HTML, B.HTML, C.HTML 과 같이 여러 페이지를 구성해야 했는데 react 는 변경된 데이터만 업데이트해준다. 또한 npm Weekly Downloads 의 수가 다른 vue, angular 보다 압도적이며 대중적이었고 커뮤니티에서 참고할 자료가 많았기 때문에 react 를 선택했다.

### 2.2.3 Project Flow



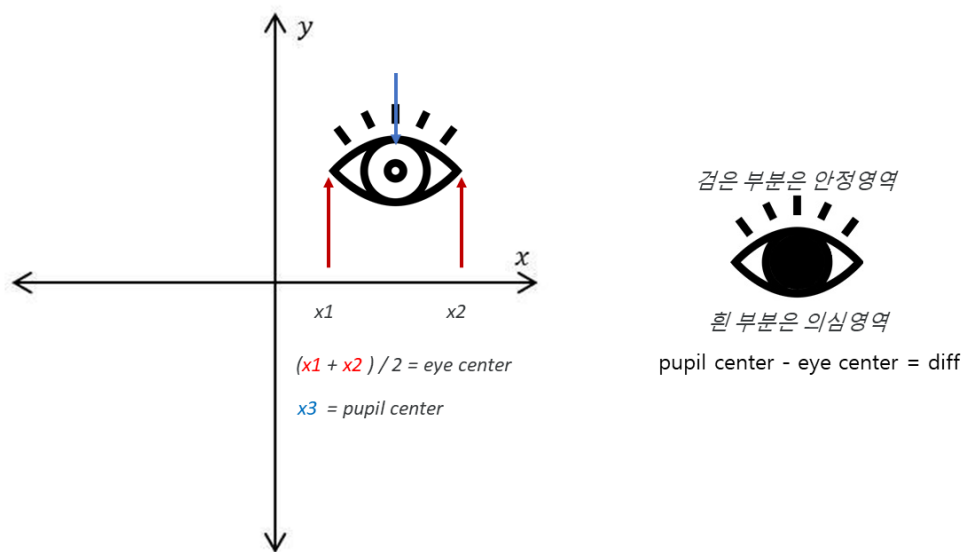
먼저 WebRTC 로부터 시작한다. WebRTC 의 모든 작업이 PeerConnection 을 거쳐 이루어진다.

PeerConnection 객체에서는 offer SDP, answer SDP 를 얻을 수 있다. Offer SDP 는 연결을 먼저 맺기를 원하는 Peer 의 SDP 를 말한다. 해당 Peer 는 자신의 Offer SDP 를 생성하여 다른 Peer 에게 전달하고 전달받은 Peer 는 Answer SDP 를 생성하여 Offer 에게 전달하여 서로의 P2P 커넥션을 맺는다. Offer 와 Answer SDP 의 공유는 WebRTC 자체에서 해결해 주지 않기 때문에 Signaling Server 로 Node.js 를 사용한다.

|   |                         |             |            |
|---|-------------------------|-------------|------------|
|  | 중간보고서                   |             |            |
|   | 프로젝트 명                  | ON감독        |            |
|   | 팀 명                     | 캡스톤 19팀     |            |
|   | Confidential Restricted | Version 1.0 | 2022-04-04 |


이후 getUserMedia API 를 통해 camera, screen 에 대한 접근 권한을 받는다. 이렇게 얻은 사용자의 webcam 은 TensorFlow 의 Facemesh 모델을 거쳐 사용자의 얼굴에 mesh 형태로 나타낼 수 있다. 여기에 구현한 cheating detection 알고리즘을 사용해 부정행위 점수를 표시하여 관리자나 감독에게 알림이 갈 수 있도록 나타내었다. 알림이나 점수로 사용자의 부정행위를 파악한 감독관은 학생에게 제재를 가할 수 있다. 또한 실시간 영상은 데이터베이스에 학생정보와 함께 저장되어 추후 확인할 수 있다.

## 2.2.4 Eye Tracking Flow



눈의 양 끝의 좌표를 잡고 이것을 x, y 각각 2로 나누어 가운데 위치를 구한다. 이것을 하나의 기준으로 만든 뒤 홍채의 위치(x, y)와 기준점(x, y)와의 차이를 저장한다. 이 차이 값이 양수로 크다면 홍채의 방향이 오른쪽으로 향했다는 것이고, 이것은 사용자가 바라보는 방향에서 왼쪽이기 때문에 홍채 방향이 왼쪽으로 움직인 것이다. 차이 값이 음수로 크다면 홍채의 방향이 오른쪽으로 향했다는 것이고, 이것은 사용자가 바라보는 방향에서 오른쪽에 해당하기 때문에 홍채가 오른쪽으로 움직인 것이다.



|  |                         |             |            |
|--|-------------------------|-------------|------------|
|  <div> 국민대학교<br/> 소프트웨어학부<br/> 캡스톤 디자인 I </div> | 중간보고서                   |             |            |
|  | 프로젝트 명                  | ON감독        |            |
|  | 팀 명                     | 캡스톤 19팀     |            |
|  | Confidential Restricted | Version 1.0 | 2022-04-04 |

### 3 수정된 연구내용 및 추진 방향

#### 3.1 수정사항

##### 3.1.1 22/03/11

(추가) 관리자와 시험 응시자를 구분하는 방식에 대하여 고민해 보아야 할 것 같다.

채팅방을 생성하면 기본적으로 관리자가 입장한 상태일 것이다. 이 상태에서 추후에 들어오는 인원은 시험 응시자로 분류하여 display 옵션에 변화를 주는 것으로 진행하여보자.

(추가) 개발을 시작하기 전에 구성되어야 할 요소를 정리해 보았으면 좋겠다.

ALL:

eye tracking

Realtime

Chat

Rooms

Notifications

Video Call

Audio Call


##### 22/03/18

(보완) 어떤 Eye Tracking 모델을 적용할 것인가?

-opencv4nodejs: 버전 호환성이 좋지 않아 오류가 너무 많아 사용하기 힘들다.

-python- DLIB 라이브러리: nodejs와 python 사이의 데이터 전달에서 문제가 발생하였다.

결론: nodejs를 사용하지 말고 python과 호환성이 좋은 Django를 사용하자. python의 라이브러리를 이용해 Eye Tracking 기능을 구현할 것이다.

|  |                         |             |            |
|--|-------------------------|-------------|------------|
|  <b>국민대학교</b><br><b>소프트웨어학부</b><br><b>캡스톤 디자인 I</b> | <b>중간보고서</b>            |             |            |
|  | <b>프로젝트 명</b>           | ON감독        |            |
|  | <b>팀 명</b>              | 캡스톤 19팀     |            |
|  | Confidential Restricted | Version 1.0 | 2022-04-04 |

### 3.1.2 22/03/25

(변경) Eye Tracking 모델에만 몰두하다 보니 실시간 처리 작업에 많은 제약이 있었다. Django에는 실시간 처리를 할 element가 없었다. 또한 Django로 실시간 화상회의를 구현하면서 커뮤니티에서 참고할 자료가 부족했다. 마지막으로 Eye Tracking을 python- DLIB 라이브러리를 적용하여 구현해 보았지만 홍채를 추적하는 작업에서 또다시 어려움을 겪으면서 Tensorflow의 Facemesh 모델을 사용하기로 한다. (Django -> Node.js), (pythonDLIB -> Tensorflow Facemesh)

(변경) 시험을 혼자서 치르지 않고, 다른 사람과 협력하는 부정행위를 어떻게 막을 수 있을까?

소리를 이용한 방법 : 화자 음성 분리를 이용하여 다양한 음성이 들리는 것으로 여러 사람이 있는 지를 파악할까? => 구현이 어려울 것 같고, 구현하더라도 높은 성능의 분리가 이뤄지기 힘들 것으로 판단한다.

ip 주소를 통한 방법 : 같은 ip를 사용한다면 같은 장소에 있는 것이 의심됨을 알리는 것이다.

=> ip 주소만 받기 때문에, 구현은 간단한 편에 속한다고 생각함. 간단한 방법으로 의심되는 학생들을 추려낼 수 있다.

### 3.1.3 22/04/01

(보완) FaceMesh를 사용하여 얼굴점을 인식하고 홍채 위치를 파악하고 있는데 이를 사용하는 치팅 방지 알고리즘을 좀 더 다듬어야 한다. -> 현재 1초마다 홍채 위치를 파악하고 부정행위인지 판별하여 Total 점수를 올려주는 방법을 이용하고 있는데 매끄럽게 다듬는 작업이 필요하다.

(수정) 추후 kurento Media server를 파악하여 장점이 많은 방향으로 진행시킬 예정이다.

|  |                         |             |            |
|--|-------------------------|-------------|------------|
|  <b>국민대학교</b><br><b>소프트웨어학부</b><br><b>캡스톤 디자인 I</b> | <b>중간보고서</b>            |             |            |
|  | <b>프로젝트 명</b>           | ON감독        |            |
|  | <b>팀 명</b>              | 캡스톤 19팀     |            |
|  | Confidential Restricted | Version 1.0 | 2022-04-04 |

## 4 향후 추진계획

### 4.1 향후 계획의 세부 내용

#### 4.1.1 로그인 기능

회원가입 및 로그인을 통해 회원 정보를 기록한다.

#### 4.1.2 신분증 대조 기능

신분증 상의 인물이 현재 시험에 응시하는 인물과 동일 인물인지 확인

#### 4.1.3 영상 저장 기능

부정행위가 의심되는 학생의 영상을 다시 시청하기 위해 영상을 저장하는 기능

#### 4.1.4 IP 위치 파악 기능

같은 공간에서 시험을 응시하는 부정행위를 막기 위한 기능

|  |                         |             |            |
|--|-------------------------|-------------|------------|
|  <b>국민대학교</b><br><b>소프트웨어학부</b><br><b>캡스톤 디자인 I</b> | <b>중간보고서</b>            |             |            |
|  | <b>프로젝트 명</b>           | ON감독        |            |
|  | <b>팀 명</b>              | 캡스톤 19팀     |            |
|  | Confidential Restricted | Version 1.0 | 2022-04-04 |

## 5 고충 및 건의사항

팀 19 건의사항 없습니다!