

1. Introduction

1.1 Purpose of system

감독관들이 모든 학생들의 화면을 일일이 확인하지 않아도 부정행위 하는 것으로 간주되는 학생들을 실시간으로 확인할 수 있다.

감독관이 시험 문제를 출제할 수 있고, 응시자들은 업로드된 시험에 응시할 수 있다.

학생들의 녹화된 시험 응시 영상은 서버에 저장되어 감독관이 재확인할 수 있다.

본 프로그램 이외의 프로그램은 실행을 차단시켜 부정행위를 방지한다.

1.2 Scope of the system

모든 시험 감독관과 응시자들이 이용할 수 있다.

언택트 시대를 맞이한 현 시점에서 주요한 역할 가능성

1.3 Objectives and success criteria of the project

회원가입자 수 1만 명, 동시 접속자 5000명 수용하는 것을 목표로 한다.

2. Current system

최근 코로나로 인해 비대면으로 수업을 하고 시험도 비대면으로 이루어지고 있다. 그러나 이러한 비대면 시험은 부정행위자를 가려내기 어렵다는 단점이 있다.

또한 시험감독관이 학생들의 ZOOM과 같은 화상 프로그램을 통해 부정행위를 막고자 하지만 학생의 수가 많아지면 학생 한 명 한 명의 시험을 감독하는 것은 쉽지 않다.

감독관의 눈을 피해 부정행위를 하는 응시자들의 사례가 많이 보고되고 있다.

Usecase model

Use case name	RegisterAccount
Participating actors	유저(응시자, 감독관)
Flow of events	<ol style="list-style-type: none">1. 유저는 “RegisterAccount” 활성화.2. 서버가 회원가입 Form을 유저에게 보여준다.3. 유저는 회원가입 Form (이름, 생년월일, 희망아이디, 희망비밀번호) 을 작성한다.4. 서버가 유저에게 본인인증 Form을 전송.5. 유저는 본인인증 Form(이름, 생년월일, 전화번호)에 입력을 한 후 인증번호 요청.6. 서버는 유저의 전화번호로 인증 번호를 전송.7. 유저는 인증번호를 입력.

	8. 서버는 회원가입 완료 Form 전송. DB에 유저 정보 저장.
Entry condition	
Exit condition	<p>유저의 정보가 서버의 DB에 정상적으로 저장된 경우.</p> <p>유저가 도중에 회원가입 창에서 벗어난 경우.</p>
Quality requirements	<p>유저의 정보는 반드시 DB에 정상적으로 안전하게 저장되어야 함.</p>

Use case name	Login
Participating actors	유저(응시자, 감독관), 서버

Flow of events	<p>1. 사용자가 “Login” 활성화</p> <p>2. 서버는 유저에게 Login Form 전송</p> <p>3. 유저는 id와 password를 Login Form에 입력한다.</p> <p>4-1. 서버는 입력 받은 id가 DB에 저장되어 있지 않거나 password가 다른 경우, 아이디 혹은 비밀번호가 잘못되었음을 알림.</p> <p>4-2. 서버는 입력 받은 id와 password가 DB에 저장되어 있는 경우, 로그인 성공을 유저에게 알림.</p>
Entry condition	
Exit condition	<p>로그인 성공.</p> <p>유저가 도중 로그인 창에서 벗어난 경우.</p>
Quality requirements	유저의 id와 password는 안전하게 서버로 전송되어야 함.

Use case name	PrepareExam
Participating actors	감독관, 서버

Flow of events	<p>1. 감독관이 “PrepareExam”을 활성화</p> <p>2. 서버는 시험 정보 입력 Form을 전송.</p> <p>3. 감독관은 해당 Form(시험의 이름, 시험 시작, 종료 시간)에 입력.</p> <p>4. 서버는 시험 문제 입력 Form 전송.</p> <p>5. 감독관은 시험 문제 입력 Form(문제 유형, 문제, 정답)에 입력.</p> <p>6. 서버는 시험에 대한 정보를 DB에 저장.</p> <p>감독관에게 시험 정보가 성공적으로 저장되었음을 알리고, 시험 코드를 보여주고, 비밀번호 설정 Form을 전송.</p> <p>7. 감독관은 비밀번호 설정 Form에 비밀번호 입력.</p> <p>8. 비밀번호를 DB에 저장.</p> <p>9. 감독관은 응시자들에게 시험 코드와 비밀번호 공지.</p>
Entry condition	감독관의 login.
Exit condition	<p>시험 정보가 DB에 성공적으로 저장.</p> <p>감독관이 시험 정보 입력 Form을 벗어났을 경우.</p>
Quality requirements	시험 정보는 안전하게 DB에 저장되어야 함.

Use case name	EnterExam
Participating actors	응시자, 서버

Flow of events	<p>1. 응시자가 “EnterExam” 활성화.</p> <p>2. 서버는 시험 코드 입력 Form 전송.</p> <p>3. 응시자가 시험코드를 입력.</p> <p>4-1. 시험코드가 DB에 없는 경우.</p> <p>존재하지 않는 시험이라고 알림.</p> <p>4-2. 시험코드가 DB에 존재하는 경우.</p> <p>시험 비밀번호 입력 Form을 응시자에게 보여줌.</p> <p>5. 시험 비밀번호 입력.</p> <p>6-1. 시험 비밀번호가 일치하는 경우.</p> <p>6-1-1. 시험 시작 시간이 지났을 경우,</p> <p>“TakeExam” 활성화.</p> <p>6-1-2. 시험 시작 시간 전,</p> <p>시험 대기화면을 띄움.</p> <p>6-1-3. 시험 종료 시간이 지났을 경우.</p> <p>시험이 종료되었음을 알림.</p> <p>6-2. 시험 비밀번호가 일치하지 않는 경우.</p> <p>비밀번호가 잘못되었음을 알림.</p>
Entry condition	응시자 로그인.
Exit condition	<p>응시자가 시험에 정상적으로 입장한 경우.</p> <p>응시자가 시험 코드, 비밀번호 입력 창에서 벗어난 경우.</p>
Quality requirements	

Use case name	TakeExam
Participating actors	응시자, 감독관, 서버, 클라이언트
Flow of events	<ol style="list-style-type: none"> 1. 서버가 “TakeExam” 활성화 2. 서버가 DB에서 해당 시험코드에 해당된 시험문제들을 응시자의 화면에 띄우고 시험 문제form을 전송 3. 응시자는 시험 문제 form에 답을 작성 4. 응시자가 입력한 답안들을 실시간으로 클라이언트 측에 저장
Entry condition	<p>응시자는 login 되어 있음.</p> <p>응시자는 EnterExam 완료</p>
Exit condition	응시자 혹은 클라이언트 프로그램이 SubmitExam 활성화.
Quality requirements	시험 도중 네트워크가 끊어졌을 경우를 대비하여 응시자의 답안을 클라이언트에 실시간 저장

Use case name	SubmitExam
Participating actors	응시자, 서버, 클라이언트

Flow of events	<p>1-1. 유저가 “SubmitExam” 활성화</p> <p>1-2. 클라이언트 프로그램이 “SubmitExam” 활성화</p> <p>2. 클라이언트가 시험 답안을 서버로 전송하고, 서버는 받은 시험 답안을 uid를 primary key로 갖는 DB 테이블에 저장.</p>
Entry condition	유저가 SubmitExam 활성화
Exit condition	답안이 성공적으로 DB에 저장된 경우.
Quality requirements	답안이 안전하게 전송되어야 함.

Use case name	Scoring
Participating actors	감독관, 서버

Flow of events	<p>1. 감독관이 “Scoring” 활성화</p> <p>2. 서버는 감독관에게 시험코드선택Form을 띄움</p> <p>3. 시험코드를 입력</p> <p>4, 해당시험코드의 시험문제 및 답안과 응시자들의 답안을 표로 정리하여 감독관에게 전송.</p> <p>채점표를 생성하여 함께 전송. (객관식은 자동채점되어 채점표에 작성된 후 감독관에게 전송됨)</p> <p>5. 감독관은 받은 표 및 파일들을 이용하여 채점표를 완성한다.</p>
Entry condition	감독관이 로그인 되어있어야 하며, 시험이 종료된 후.
Exit condition	감독관이 파일을 전송 받음.
Quality requirements	감독관에게 해당 파일들이 안전하게 전송되어야 함

Use case name	Recording
Participating actors	클라이언트, 서버

Flow of events	<p>1. 클라이언트가 “Recording” 활성화</p> <p>2. 클라이언트는 실시간으로 응시자의 웹 캠 화면 영상을 서버로 전송, “CheckCheating” 활성화 시킴.</p> <p>3. 서버는 받은 영상을 유저 id 별로 storage에 저장.</p> <p>저장된 경로를 DB에 저장.</p>
Entry condition	응시자가 TakeExam 활성화.
Exit condition	SubmitExam 활성화.
Quality requirements	영상 데이터는 일부 누락되어도 큰 문제가 없으므로, TCP 보다는 UDP를 이용하여 실시간 전송.

Use case name	CheckCheating
Participating actors	클라이언트, 서버

Flow of events	<p>1. 클라이언트가 “CheckCheating” 활성화 시킴.</p> <p>2. “Recording”에서 받은 영상을 아이트래킹 모델에 input으로 입력함.</p> <p>3. 아이트래킹 모델의 결과 값으로 부정행위 판별.</p> <p>4. 부정행위로 의심될 경우 클라이언트는 서버에게 부정행위 정보 전송.</p>
Entry condition	TakeExam, Recording 동작 중.
Exit condition	SubmitExam 활성화.
Quality requirements	아이트래킹 모델은 일정 수준 이상의 정확성을 가져야 함.

Use case name	NotifyCheating
Participating actors	감독관, 서버

Flow of events	<p>1. 서버가 “NotifyCheating” 활성화.</p> <p>2. 서버가 감독관 클라이언트에게 부정행위 정보(응시자, 시선 각도, 횡수) 전송.</p>
Entry condition	TakeExam, Recording, CheckCheating 동작 중.
Exit condition	SubmitExam 활성화.
Quality requirements	

Use case name	Supervising
Participating actors	감독관, 클라이언트, 서버

Flow of events	<p>1. 서버 “Supervising” 활성화</p> <p>2. 감독관 클라이언트는 서버에서 받은 부정행위정보들을 저장하다.</p> <p>3. 감독관 클라이언트는 10초마다 전송 받은 정보를 토대로 화면상태를 업데이트한다. 이때, 화면 업데이트는 부정행위 확률이 높은 학생의 화면을 상위노출하는 방식으로 이루어진다. 또한, 부정행위관련 정보도 화면에 같이 표시한다.</p> <p>4. 감독관은 전체학생들의 화면을 확인할 수 있되, 부정행위의 확률이 높은 학생들의 화면을 좀 더 크게 확인 가능하다.</p>
Entry condition	NotifyCheating 작동
Exit condition	SubmitExam 활성화.
Quality requirements	

Use case name	Chatting
Participating actors	감독관, 응시자, 클라이언트, 서버

Flow of events	<p>1. 유저가 채팅내용 및 수신자정보를 입력한 후, 전송버튼을 눌러 “Chatting”을 활성화 시켜 서버로 해당내용을 전송함</p> <p>2. 서버는 전달받은 내용을 수신자에게 전송</p>
Entry condition	TakeExam 동작 중.
Exit condition	SubmitExam 활성화.
Quality requirements	<p>-시험 도중 응시자와 감독관의 소통이 필요한 경우 대비</p> <p>Ex1)응시자가 시험관련 질문이 있는 경우</p> <p>Ex2)감독관이 응시자가 부정행위 중이라고 의심이 될 때 카메라를 돌려보라는 등의 요구를 해야 하는 경우</p> <p>-실시간</p>

Use case name	CheckRecord
Participating actors	감독관, 클라이언트, 서버

Flow of events	<p>1. 감독관이 “CheckRecord” 활성화</p> <p>2. 서버는 감독관에게 관련 form전송</p> <p>3. 감독관은 form에 확인하고 싶은 시험코드와 해당학생의 아이디를 입력</p> <p>4. 서버는 감독관이 확인하고자 하는 해당비디오를 감독관에게 전송</p> <p>5, 감독관은 해당 영상을 다운로드 받는다.</p>
Entry condition	시험이 종료된 후
Exit condition	감독관이 해당영상 다운로드를 성공적으로 끝냄
Quality requirements	