



snowcode

과제 채점 플랫폼



FE: 백수민, 하지민

BE: 주아정

1. 서비스 기획 배경

소프트웨어의 이해는 소프트웨어학부 1학년 전공필수 과목으로
2026년 1학기 기준 5개 분반 · 총 259명이 수강하는 대형 강의입니다



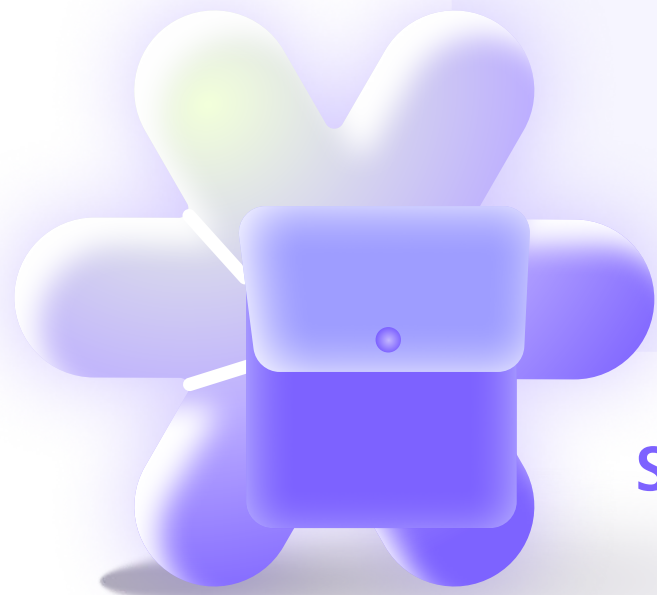
1. 서비스 기획 배경

복잡한 제출 프로세스: 소스 파일을 별도로 정리하고, 실행 결과를 캡처해 보고서에 첨부해야 해요.

잘은 제출 오류: 파일 누락이나 형식 오류로 제출 과정에서 실수가 생기기 쉬워요.

긴 채점 대기 시간: 제출 후 결과를 받기까지 오래 기다려야 해요.

환경 불일치 문제: 로컬에서 잘 되던 코드가 채점 환경에서 오류가 나기도 해요.

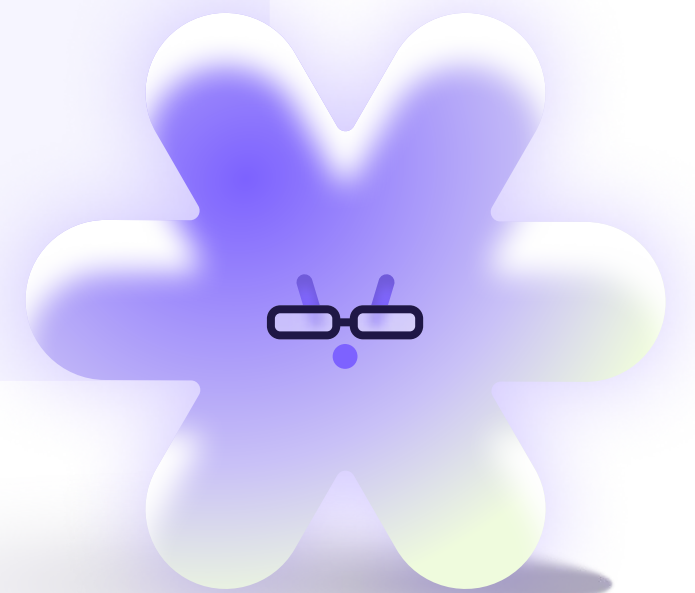


STUDENT PAIN POINTS

수동 채점 부담: 제출물을 하나씩 다운로드해서 직접 채점해야 해요.

비효율적인 문의 처리: Q&A 게시판이 있어도 대부분의 질문이 이메일로 들어와요.

환경 재현의 어려움: 학생마다 다른 라이브러리 버전 때문에 실행 환경을 맞추기가 어려워요.

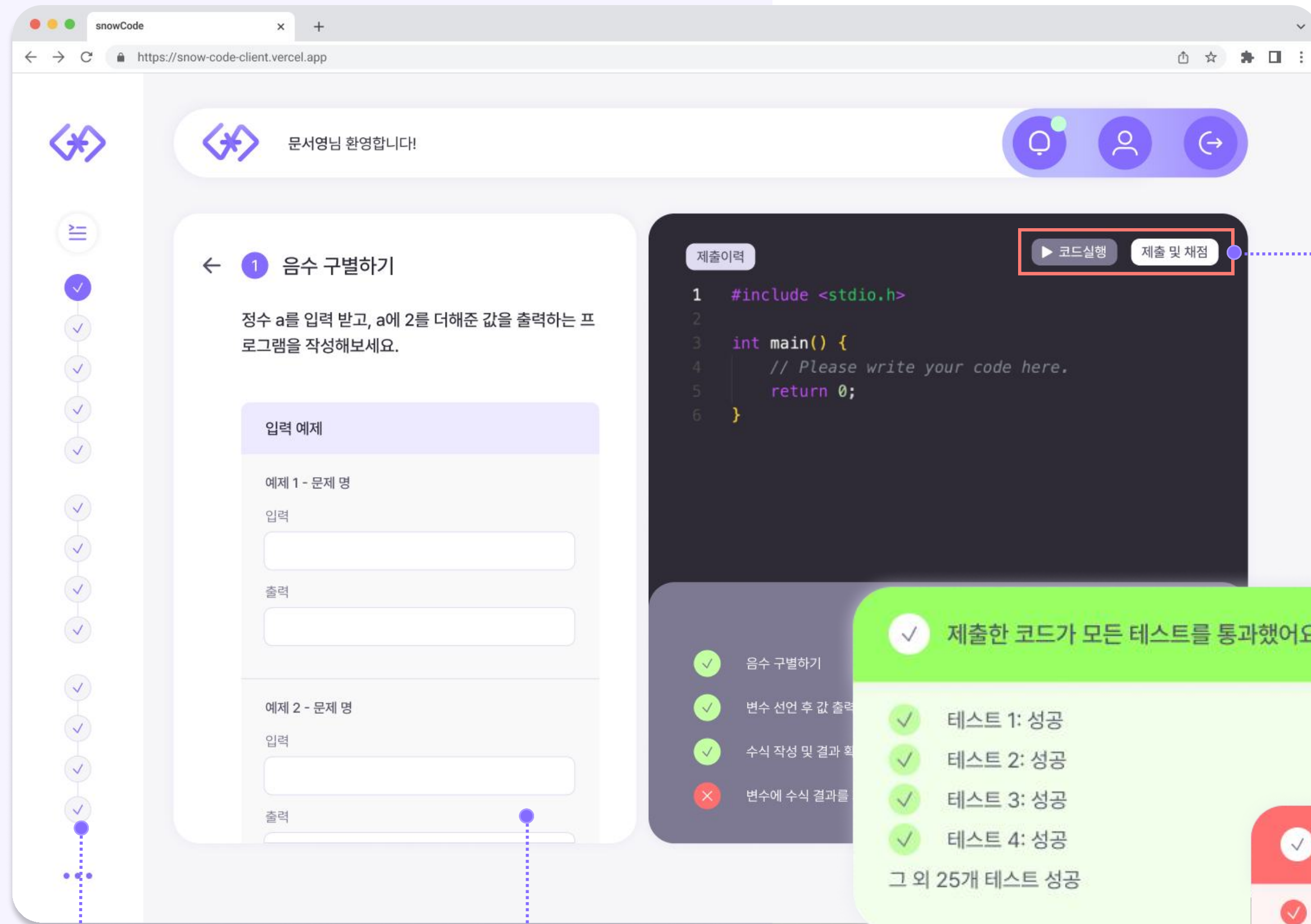


TA PAIN POINTS



2. 주요 기능

코드 제출 · 코드 실행



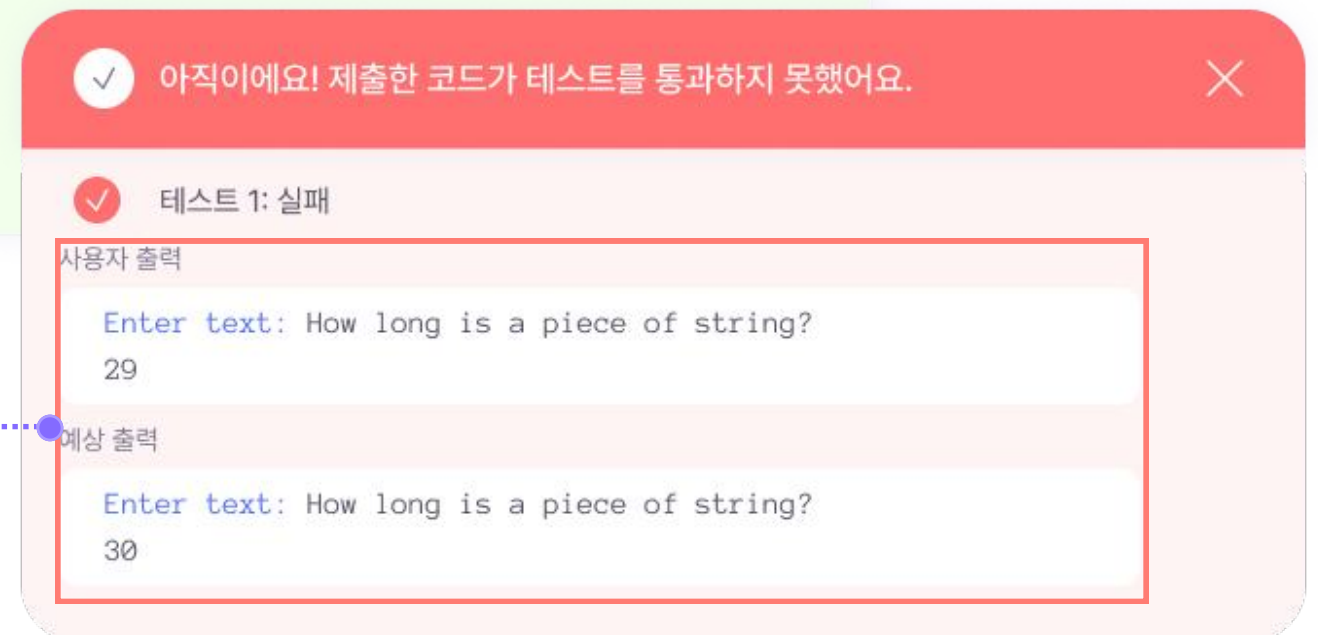
웹 IDE 환경에서 코드 작성 · 실행 ·
제출 · 채점 원스톱 처리

제출 후 테스트 케이스별
성공 여부 즉시 확인

문제별 제출 현황 확인 및
문제 간 빠른 이동 지원

공개 테스트 케이스 기반
입출력 예제 제공

공개 테스트 케이스의
예상 출력과 실제 출력 비교로
오답 원인 파악 가능



2. 주요 기능

강좌 관리

체계적인 단위 및 문제 관리

단원을 구분하여 전체적인 커리큘럼 구조 쉽게 관리 가능

강의 개설

강의 명

강의 분반을 입력하세요

연도

강의 연도를 선택하세요

학기

강의 학기를 선택하세요

강의 소개

강의 소개를 입력하세요

강의 공유

공유자를 입력하세요

대표 관리자를 선택하세요

취소

저장

단원 보기

2. 문제

변수와 수식

3. 문제

판단과 선택

2. 문제

반복

4. 문제

파일 입출력

2. 문제

오류와 예외처리

+ 단위 추가

1. 단위

제목

제목을 입력하세요

공개일

날짜를 선택하세요

마감일

날짜를 선택하세요

문제 등록

:: 음수 구별하기

:: 음수 구별하기

+ 문제 연결

취소

저장

효율적인 강좌 개설 프로세스

통합 입력 폼: 강의명, 분반, 연도, 학기 등
강좌의 기본 정보 한 눈에 파악 가능

계층적 단위 구성

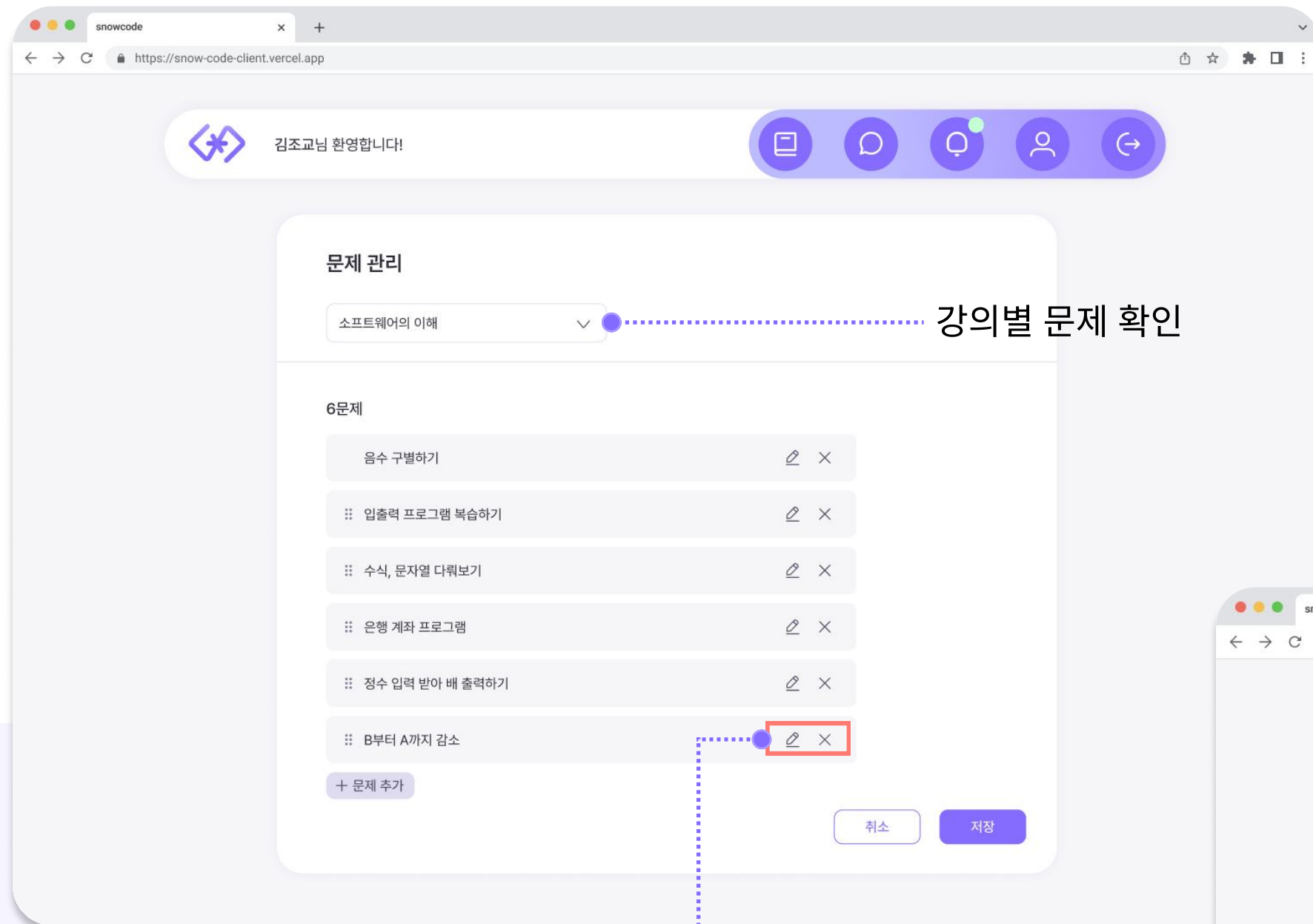
'단원 추가' 버튼을 통해
커리큘럼에 맞춘
학습 단위 무제한으로 생성 가능



2. 주요 기능

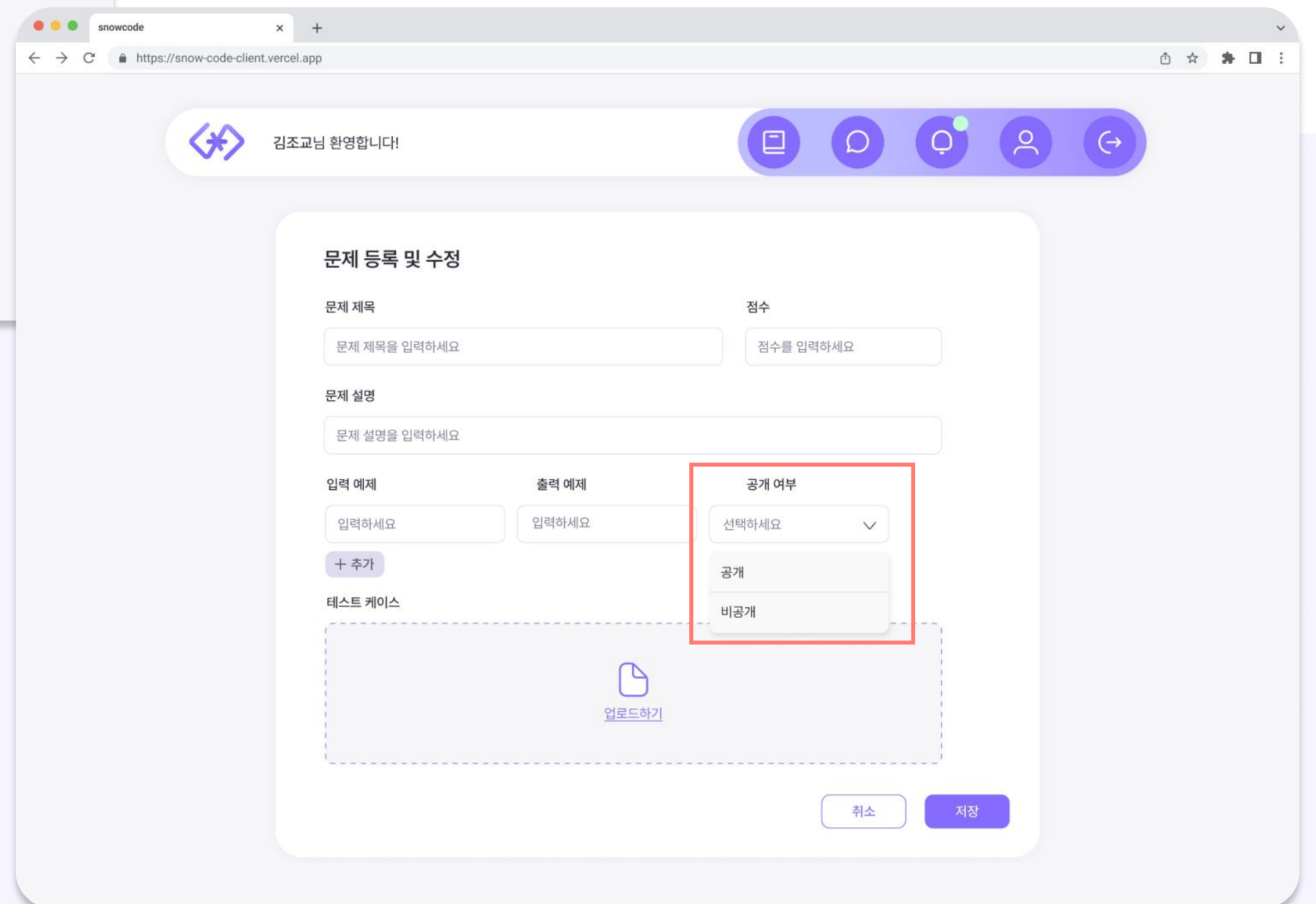
과제 출제 및 관리

1. 문제 관리 전용 페이지에서 문제를 강좌와 분리하여 **독립적으로 관리**
2. 문제 설명 · 제목 · 배점 · 테스트 케이스 등록
3. 한번 만든 문제를 **여러 강좌 · 단원에 재사용**해 출제 준비 시간 단축
4. 원하는 단원에 문제를 자유롭게 배치



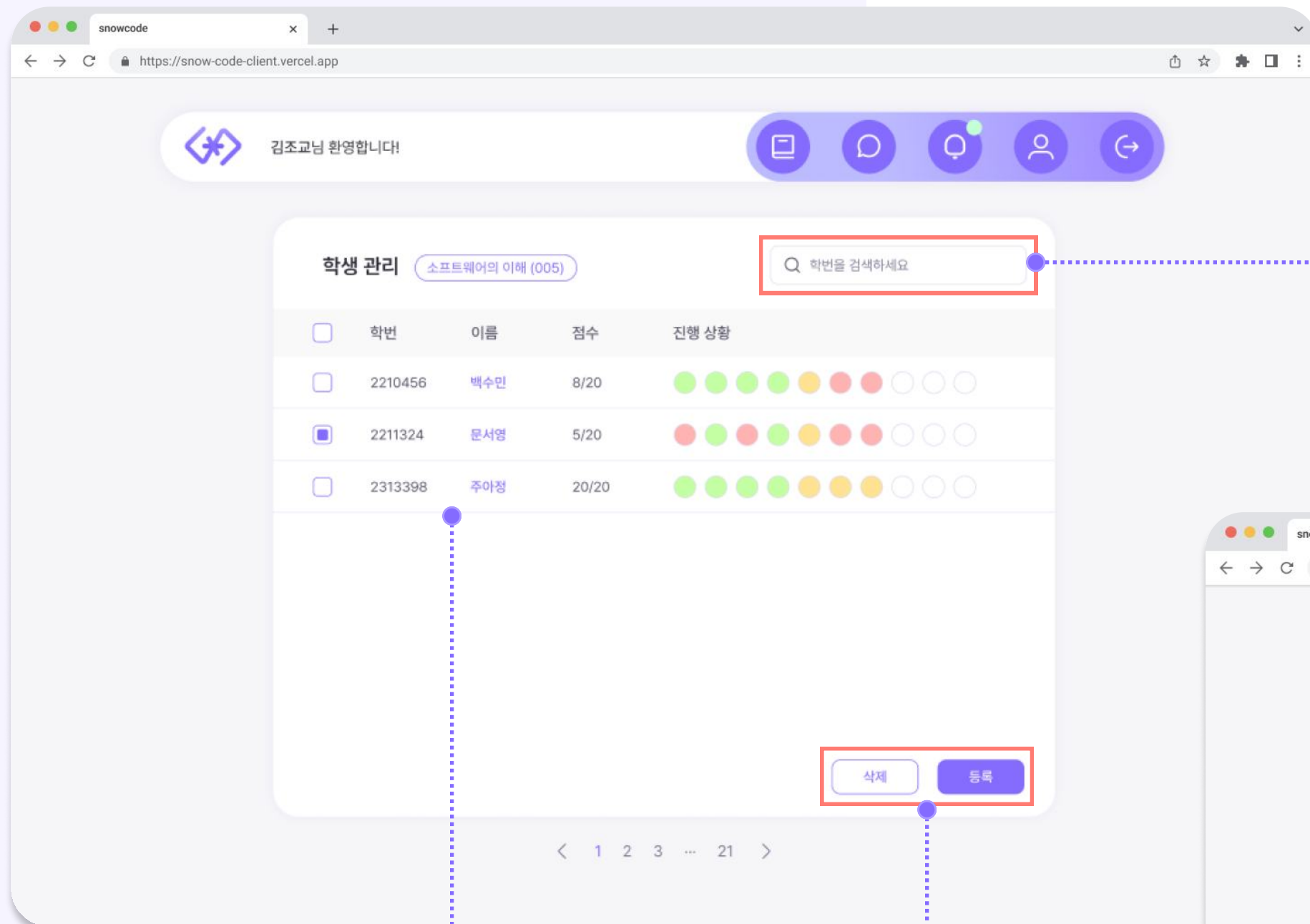
문제 수정 / 삭제

테스트케이스
공개 여부 설정



2. 주요 기능

학생 관리 및 조회



빠른 학생 검색

학번 검색으로 원하는 학생 조회

학습 현황 상세 조회

학생별 과제 제출 여부 · 점수 · 제출 코드 조회로
개별 학습 현황 모니터링 가능

전체 수강생 현황 파악

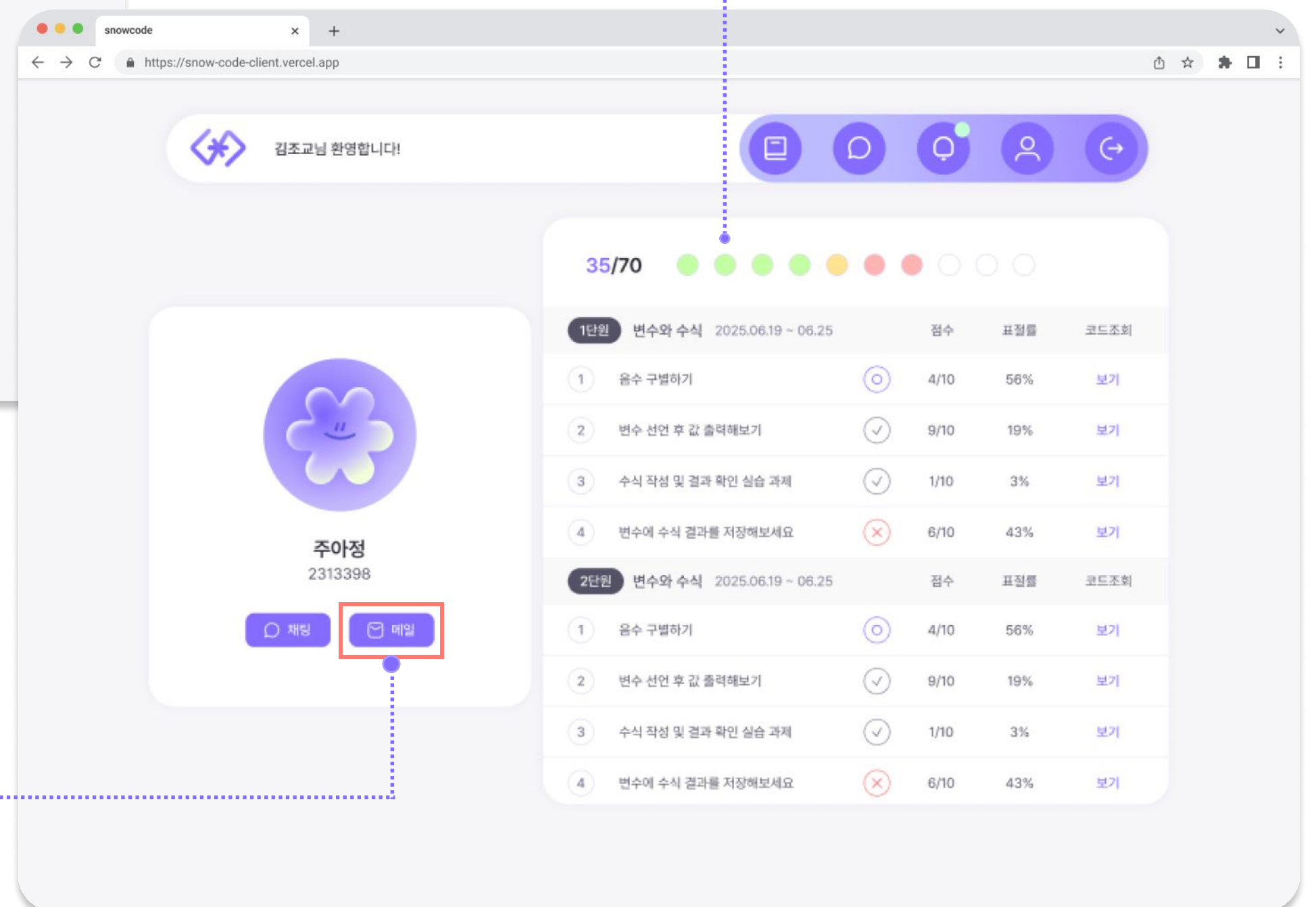
수강생 학번 · 이름 · 점수 · 단원별
진행 상황을 한 페이지에서 확인

수강생 관리

학생 등록 · 삭제 지원

이메일 연락

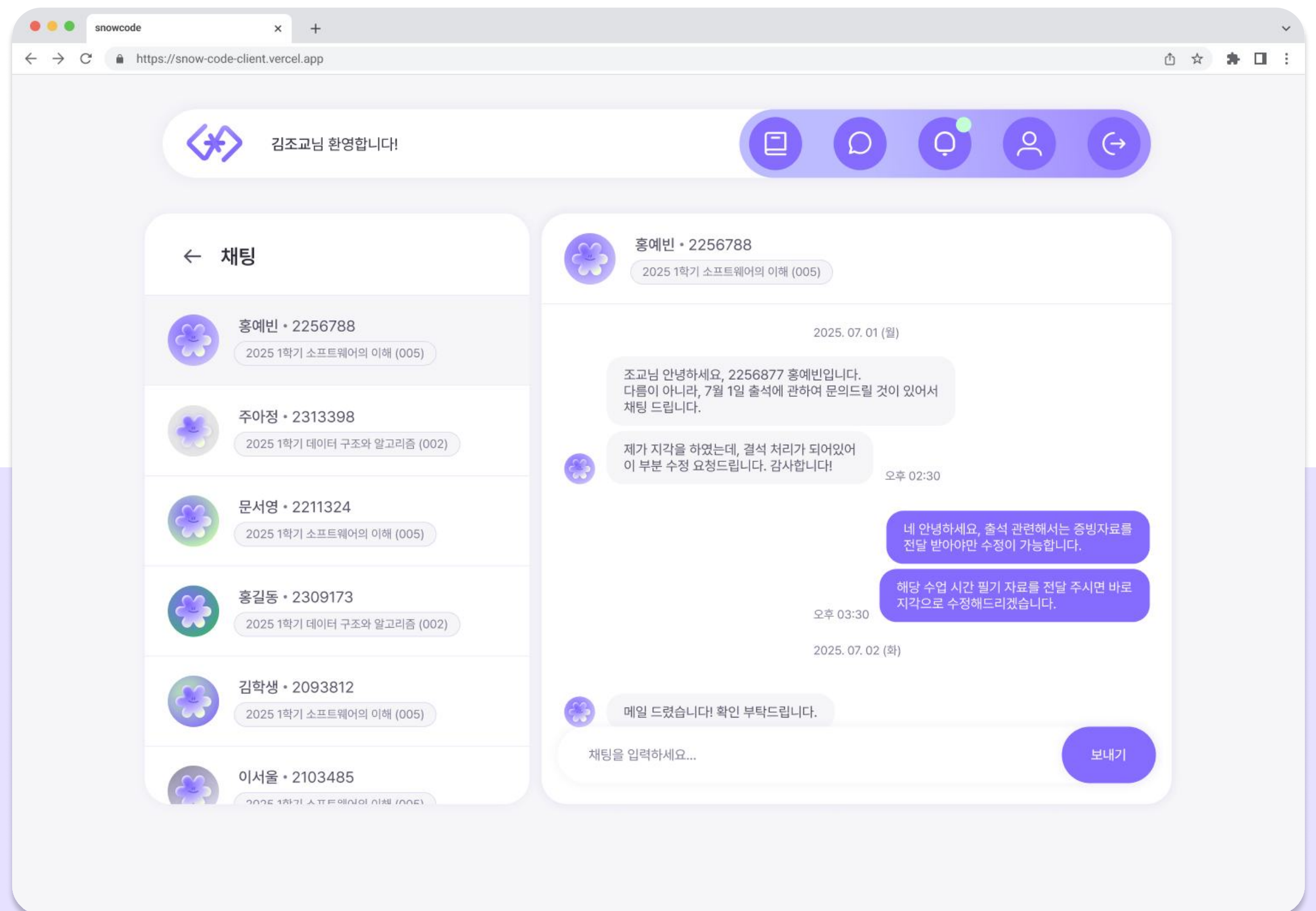
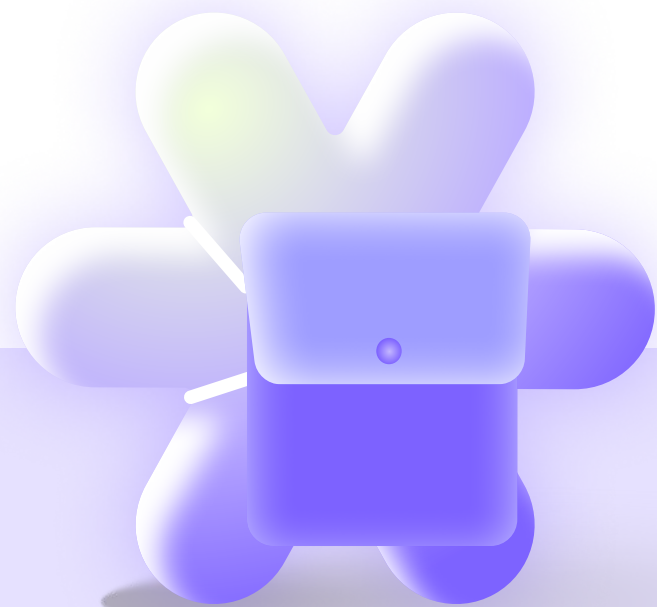
기본 메일 앱과 연동되어 해당
학생에게 직접 이메일 연락 가
능



2. 주요 기능

실시간 채팅

1. 학생 ↔ 조교 간 과제 관련 **실시간 1:1 문의**로 이메일 문의 없이 빠른 소통 가능
2. 과제 제출 페이지에서 바로 문의 가능
3. **코드 전송 버튼**으로 현재 작성 중인 코드를 즉시 전달 · 코드 블록으로 가독성 있게 표시



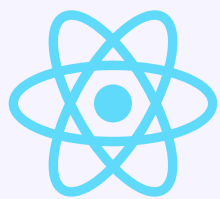
3. 기술 스택

프론트엔드 기술 스택

언어 및 프레임워크

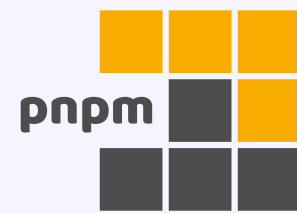


타입스크립트
TypeScript



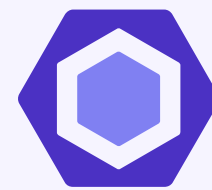
리액트
React

패키지 관리



피엔피엠
PNPM

Lint



이에스린트
ESLint

데이터 패칭



텐스택쿼리
Tanstack Query

유효성 검사



조드
Zod

스타일



테일윈드 CSS
Tailwind CSS

실시간 통신



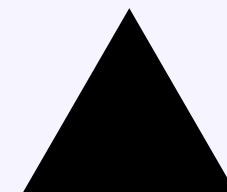
스톰프
STOMP

환경



비트
Vite

배포



버셀
Vercel



3. 기술 스택

백엔드 기술 스택

언어 및 프레임워크



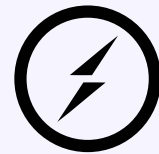
자바
Java17



스프링부트
SpringBoot



스프링 데이터 JPA
Spring Data JPA



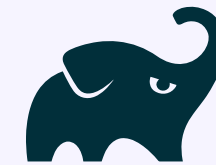
웹소켓
WebSocket.io

인증/인가

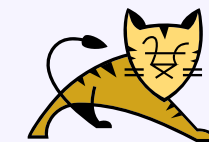


JWT
JWT

빌드 및 런타임



그레이들
gradle



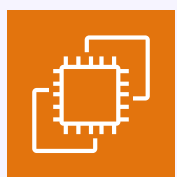
아파치 톰캣
Apache Tomcat

데이터베이스



마이애스큐엘
MySQL

클라우드

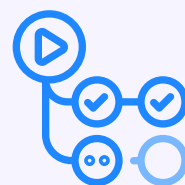


AWS
EC2



AWS
RDS

인프라 및 환경



깃허브액션
GitHub-Actions



도커
Docker

NGINX

엔진엑스
Nginx



우분투
Ubuntu 22.04 LTS



구글 클라우드
GCP

모니터링 및 테스트



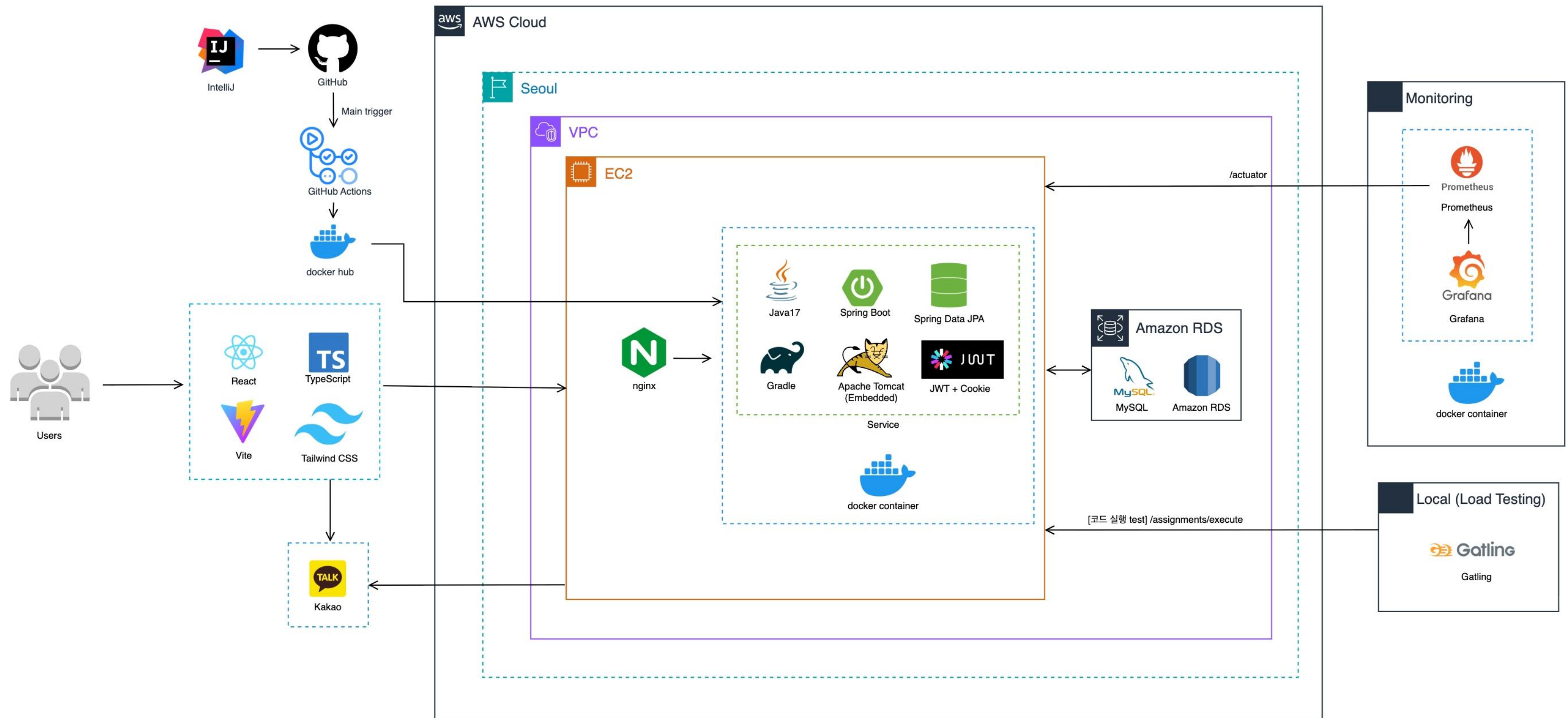
프로메테우스
Prometheus



개틀링
Gatling



4. 시스템 구조



4. 시스템 구조

실제 시나리오 성능 테스트

1. 테스트 환경 및 시나리오

- vCPU 2코어
- 메모리 2GB

- 초당 4명 유입 · 최대 동시 접속 720명
- 80명 × 3개 분반 × 3개 강의 — 마감 직전 동시 제출 최악의 상황 가정
- 재현 흐름 : 접속 유지 → 코드 실행 → 연결 종료

2. 적용 기법

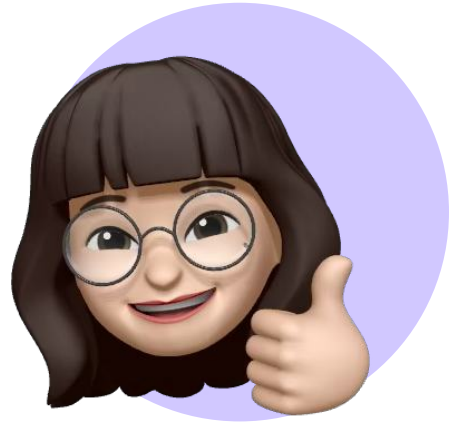
- Multi-Poller 도입
- 비동기 콜백 구조 전환
- Nginx worker_connections 튜닝
- Async-Blocking 구조 제거

3. 최적화 결과

	BEFORE	AFTER	개선율
최대 응답 지연	1,220ms	223ms	▼ 81.7%
스레드 생성 수	4 c/s	0.16 c/s	▼ 96%

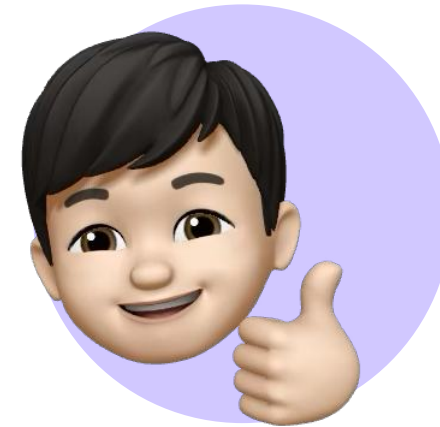


5. 기대 효과



학생 입장

- 웹 IDE에서 코드 작성부터 제출까지 원스톱 처리
- 제출 즉시 자동 채점 결과 확인
- 실행 환경이 통일되어 환경 설정 문제 없이 안정적인 코드 실행 가능
- 오답 결과를 즉시 확인하며 반복 학습 및 자기 주도 학습 강화



교수/조교 입장

- 테스트 케이스 기반 자동 채점으로 수동 채점 부담 제거
- 제출된 코드 - 소스코드 다운로드 및 별도 보고서 검토 불필요
- 통일된 실행 환경을 통해 라이브러리 설치 및 환경 설정 부담 해소

과제 출제 → 제출 → 채점 → 피드백까지 단일 플랫폼에서 완결



snowcode는 안정적인 실행 인프라를 기반으로,
정밀한 평가 검증 → 운영 효율화 → 맞춤형 학습 분석 데이터 플랫폼으로 진화합니다.

1

안전한 코드 실행 인프라

- 샌드박스 환경
- C, C++, Java, Python 등 다양한 언어 지원
- 안정성 확보
- 표준 입력(stdin) 지원

2

고도화된 평가 및 검증

- 표절 검사 지원
- 테스트 케이스 관리
- 버전 관리

3

효율적인 운영 프로세스

- 권한 분리 시스템
- 학생 일괄 등록
- 강의별/학생별 실습 현황 파악 위한 대시보드

4

통합 교육 플랫폼 확장

- 학습 분석 확장
- 지식 아카이빙
- 기관 커스텀





snowcode

과제 채점 플랫폼

감사합니다

