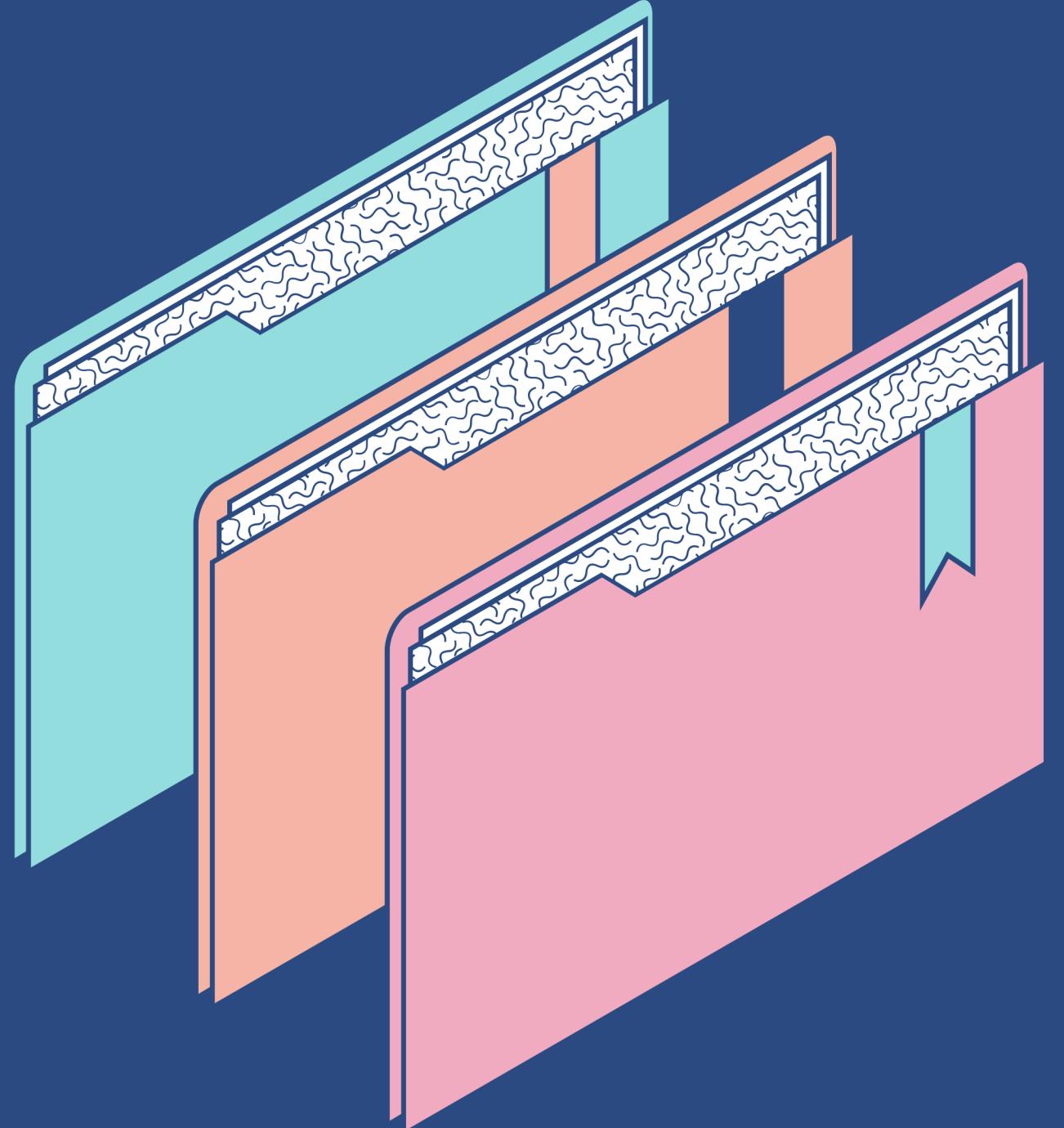




2023SPRING 41CLASS TEAM3

개발자를 위한 내비게이션: DevNavi

권민성, 서주원, 안서현, 안정민, 이장엽, 한수현



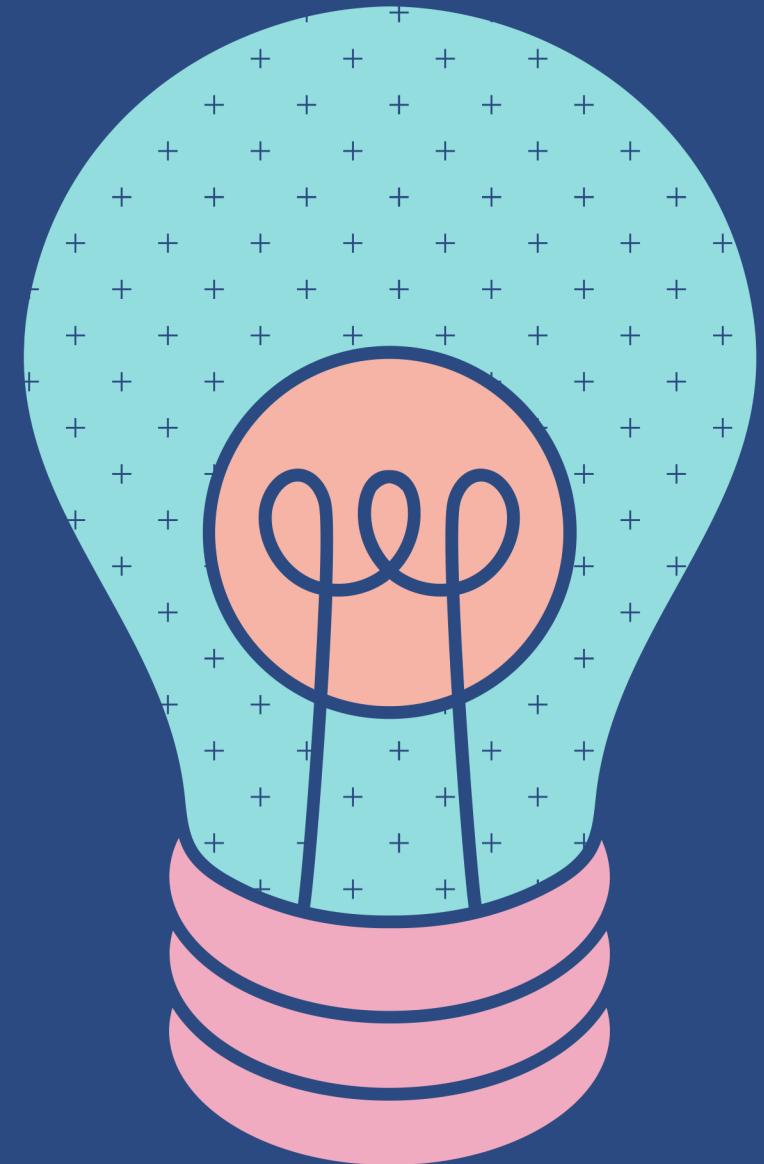
Project Overview

"코딩 테스트 연습을 위한 웹페이지"

- 사용자가 코딩 테스트를 직접 연습해보고,
- 테스트에서 작성한 코드의 가독성이나 효율성 등을 AI를 활용해 점검하면서,
- AI가 질문하는 프로그래밍 지식에 관해 답변을 하는 가상 면접을 진행할 수 있는 플랫폼

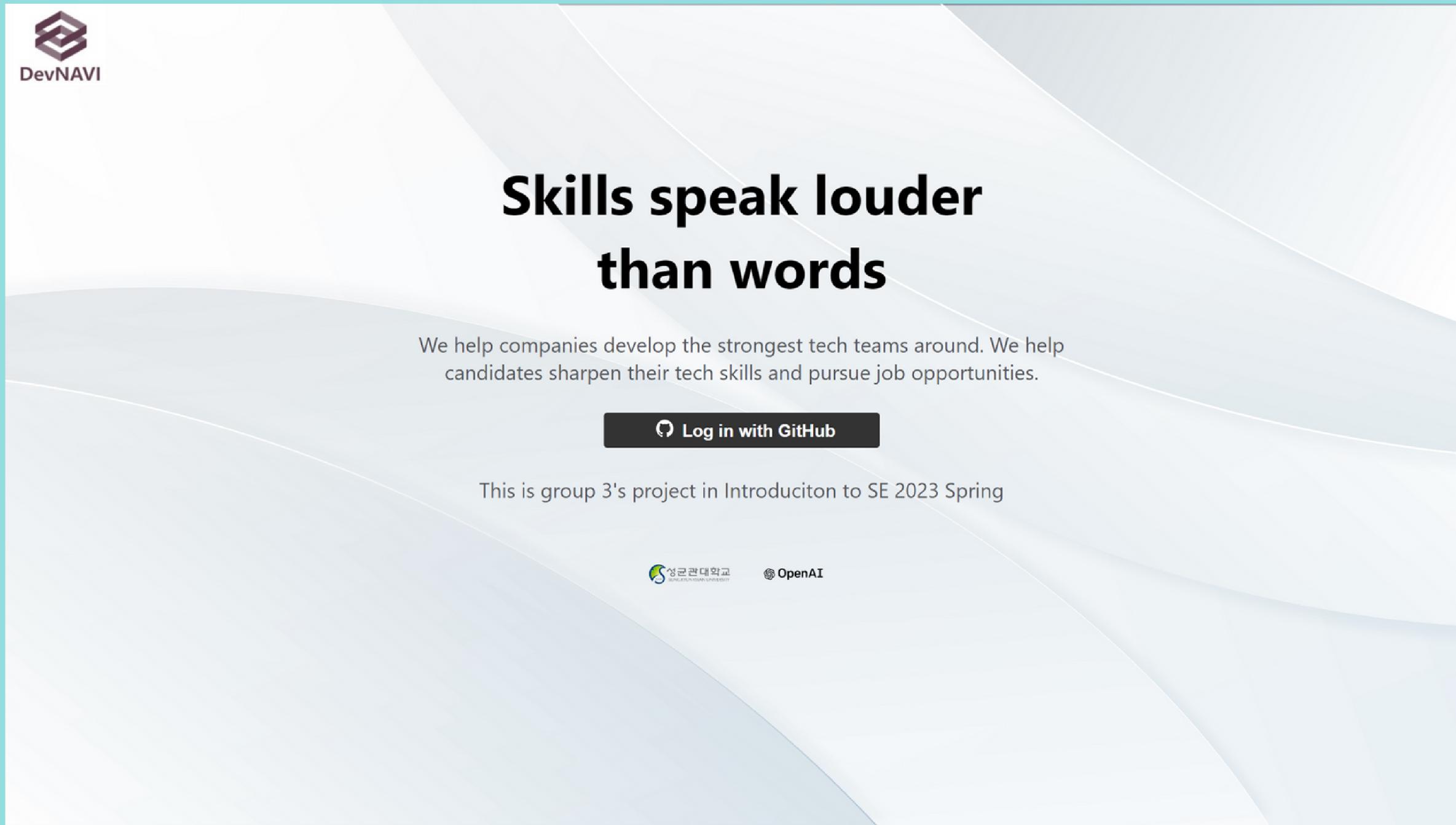
Implementation

- LOGIN PAGE
- MAIN PAGE
- PS PAGE
- FEEDBACK PAGE
- CI/CD



Login Page

- GitHub 계정을 통해 로그인
- OAuth 2.0 프로토콜을 통해 유저 인증
- JWT를 통한 세션 유지



The screenshot shows the DevNAVI login page. In the top left corner, there is a purple logo consisting of three stylized squares and the text "DevNAVI". The main title "Skills speak louder than words" is centered in large, bold, black font. Below the title, a subtitle reads: "We help companies develop the strongest tech teams around. We help candidates sharpen their tech skills and pursue job opportunities." A "Log in with GitHub" button is located below the subtitle. At the bottom of the page, there is a note: "This is group 3's project in Introduciton to SE 2023 Spring". Logos for Sungkyunkwan University and OpenAI are at the very bottom.

Skills speak louder
than words

We help companies develop the strongest tech teams around. We help candidates sharpen their tech skills and pursue job opportunities.

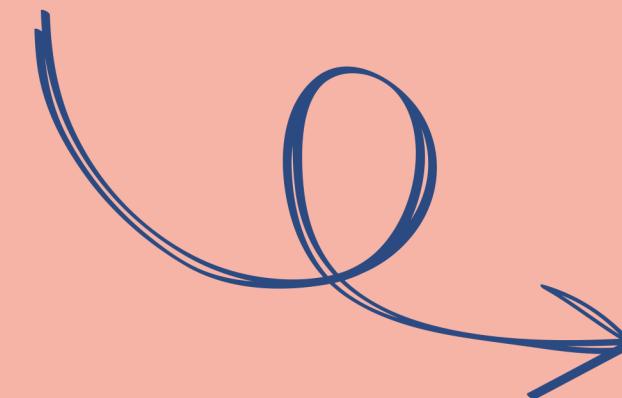
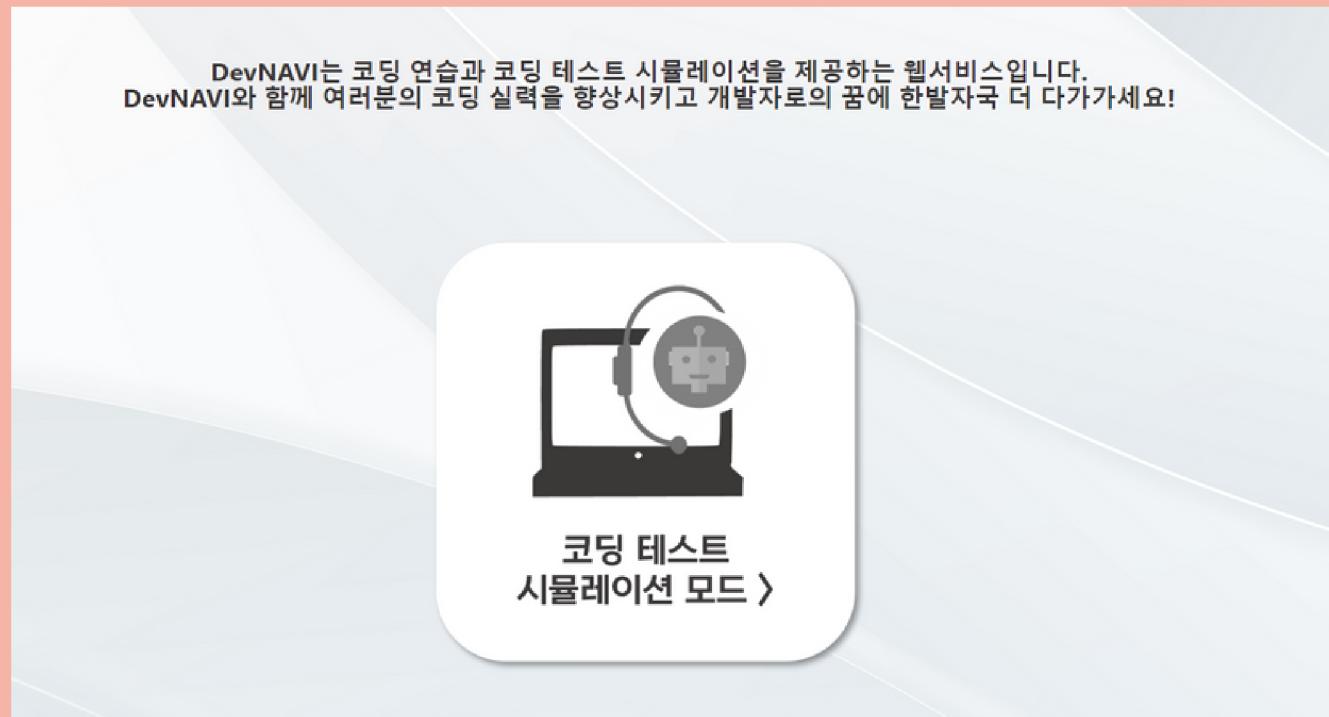
Log in with GitHub

This is group 3's project in Introduciton to SE 2023 Spring

 성균관대학교
SUNGKYUNKWAN UNIVERSITY

 OpenAI

Main Page



클릭을 통해
코딩 테스트 시뮬레이션 모드로 진입

DevNAVI는 코딩 연습과 코딩 테스트 시뮬레이션을 제공하는 웹서비스입니다.
DevNAVI와 함께 여러분의 코딩 실력을 향상시키고 개발자로의 꿈에 한발자국 더 다가가세요!

코딩 테스트 시뮬레이션

난이도 1 단계 ▾

문제 수 1 개 ▾

예상 소요시간은 10분입니다

내 코딩테스트 치러 가기

코딩 테스트 시뮬레이션

정해진 범위 내에서 난이도와 문제 수 설정

| | |
|---------|--------|
| 난이도 | 2 단계 |
| 문제 수 | 4 개 |
| 예상 소요시간 | 80분입니다 |

내 코딩테스트 치러 가기

난이도 2 단계 ▾

문제 수 4 개 ▾

예상 소요시간은 80분입니다

니다

내 코딩테스트 치러 가기

설정에 따른 예상 소요시간 표시

PS Page

- 앞서 설정한 조건에 따라 문제 구성
- 해당 페이지에서 사용자가 문제 풀이를 진행

The screenshot shows a DevNAVI platform interface for a programming challenge. On the left, a sidebar lists four problems: 문제 1 (모음 사전), 문제 2, 문제 3, and 문제 4. The selected problem is 문제 1, titled "모음 사전". The task description states: "사전에 알파벳 모음 'A', 'E', 'I', 'O', 'U'만을 사용하여 만들 수 있는, 길이 5 이하의 모든 단어가 수록되어 있습니다. 사전에서 첫 번째 단어는 "A"이고, 그다음은 "AA"이며, 마지막 단어는 "UUUUU"입니다." Below this, there is a note: "단어 하나 word가 매개변수로 주어질 때, 이 단어가 사전에서 몇 번째 단어인지 return 하도록 solution 함수를 완성해주세요." A "제한사항" section lists: "- word의 길이는 1 이상 5 이하입니다." and "- word는 알파벳 대문자 'A', 'E', 'I', 'O', 'U'로만 이루어져 있습니다." On the right, the main workspace shows a dark-themed code editor with the following C code:

```
1 #include <stdio.h>
2
3
4 int main(void) {
5     int a;
6     scanf("%d", &a);
7
8     return 0;
9 }
```

At the top of the workspace, settings are shown: Theme (dark), Language (C), and Time (남은 시간: 00:01:30). At the bottom of the workspace, there is a "Run code result" area and a "RUN CODE" button. A "제출" (Submit) button is located at the very bottom right.

문제 1

문제 2

문제 3

문제 4

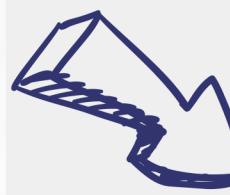
테스트 코드 실행 및
테스트 결과 표시

메뉴 바 버튼을 통해 문제 이동

The screenshot shows a code editor interface. At the top, there's a navigation bar with 'Theme' set to 'dark', 'Language' set to 'C', and a timer showing '남은 시간: 00:00:00'. On the left, a sidebar lists four problems: '문제 1', '문제 2', '문제 3', and '문제 4'. The main area contains a dark-themed code editor with the following C code:

```
1 #include <stdio.h>
2
3
4 int main(void) {
5     int a;
6     scanf("%d", &a);
7
8     return 0;
9 }
```

At the bottom, there's a results panel titled 'Run code result' with a 'RUN CODE' button and a '제출' (Submit) button.



에디터 테마, 코딩 언어 선택
남은 시간 표시



코드 제출

Feedback Page

- chat GPT를 활용한 코드 피드백
- 가독성, 시간 복잡도 측면에서 분석

The screenshot shows a feedback page from DevNAVI. On the left, there's a sidebar with navigation links: Problem 1 (selected), Problem 2, Problem 3, Problem 4, Problem 5, 분석 (Analysis), and 피드백 (Feedback). The main area displays a code snippet and its analysis:

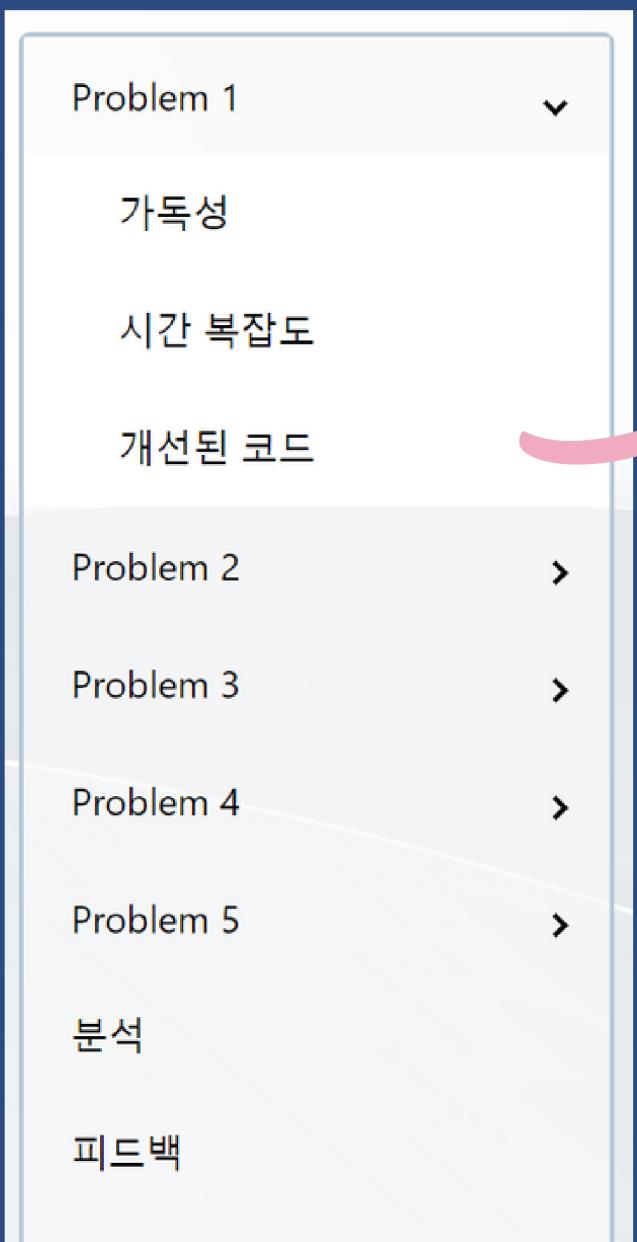
분석

코드는 상당히 짧고 간결하게 작성되어 있습니다. 변수 이름도 적절하게 지어져 있고, 주석이나 문서화가 없어도 코드의 의도를 파악하기 쉽습니다.

하지만, 코드 내에서 사용된 변수명 dic1과 dic2는 명확한 의미를 전달하지 않습니다. 딕셔너리를 사용하는 목적과 구조를 설명하는 주석이 추가되면 더욱 명확해질 것입니다. 또한, 함수 이름 solution도 어떤 문제를 해결하는지 알려주지 않으므로 함수 이름을 보완하는 것이 좋습니다.

메뉴 바를 통해

원하는 내용 확인 가능



The image shows a code editor with two Python functions. The first function, `solution`, takes a word as input and calculates a score based on two dictionaries. The second function, `calculate_score`, takes a word and calculates a score using a score table and a factor of 5 raised to the power of the index.

```
1 def solution(word):
2     dic1 = {'A': 0, 'E': 1, 'I': 2, 'O': 3, 'U': 4}
3     dic2 = {0: 781, 1: 156, 2: 31, 3: 6, 4: 1}
4     answer = 0
5     for i in range(len(word)):
6         answer += dic1[word[i]]*dic2[i]+1
7
8    return answer

1 def calculate_score(word):
2     score_table = {'A': 0, 'E': 1, 'I': 2, 'O': 3, 'U': 4}
3     answer = 0
4     for i, letter in enumerate(word):
5         score = score_table[letter]
6         factor = pow(5, i)
7         contribution = (score * factor) + 1
8         answer += contribution
9
10    return answer
```

The image shows a code editor with two Python functions. The first function, `solution`, takes a word as input and calculates a score based on two dictionaries. The second function, `calculate_score`, takes a word and calculates a score using a score table and a factor of 5 raised to the power of the index.

```
1 def solution(word):
2     dic1 = {'A': 0, 'E': 1, 'I': 2, 'O': 3, 'U': 4}
3     dic2 = {0: 781, 1: 156, 2: 31, 3: 6, 4: 1}
4     answer = 0
5     for i in range(len(word)):
6         answer += dic1[word[i]]*dic2[i]+1
7
8    return answer

1 def calculate_score(word):
2     score_table = {'A': 0, 'E': 1, 'I': 2, 'O': 3, 'U': 4}
3     answer = 0
4     for i, letter in enumerate(word):
5         score = score_table[letter]
6         factor = pow(5, i)
7         contribution = (score * factor) + 1
8         answer += contribution
9
10    return answer
```

Interview Page

가상 면접 진행

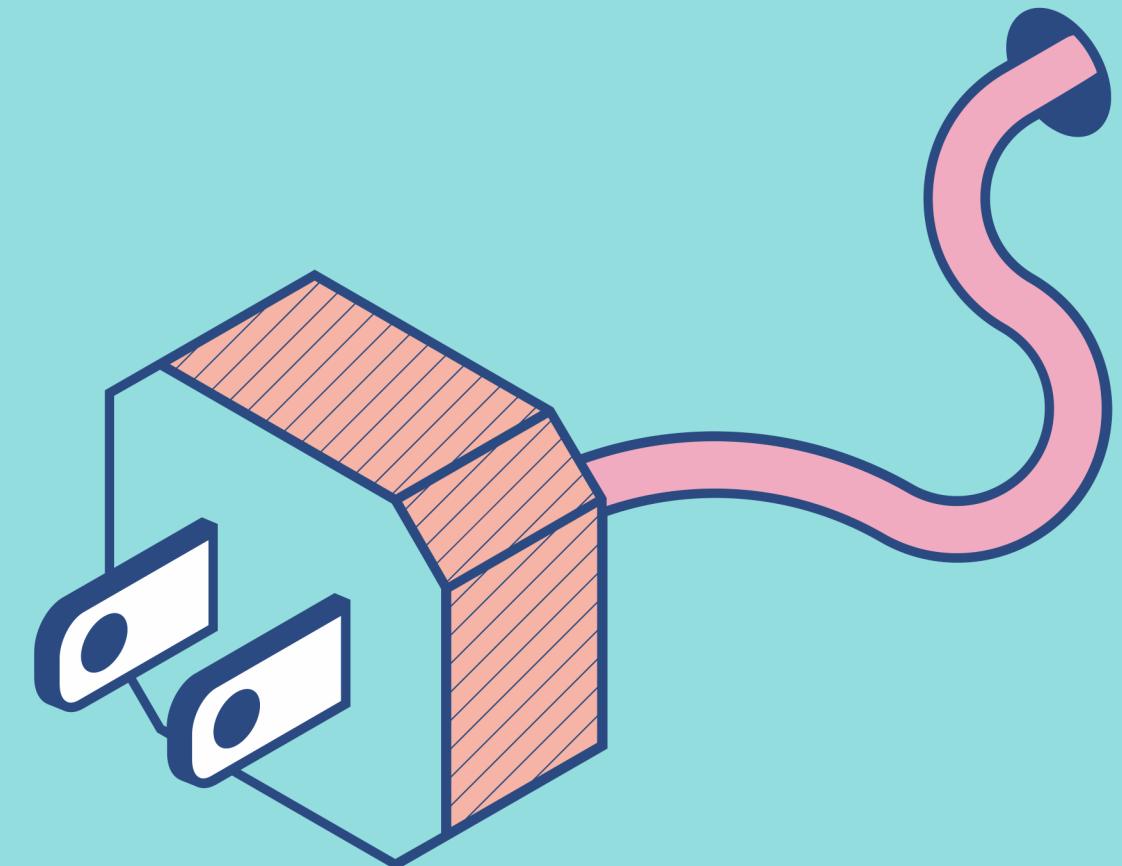
사용자는 질문에 대한 답을 입력

AI를 활용하여 답변 평가

The image shows a mockup of the DevNAVI Interview Page. At the top left is the DevNAVI logo, which consists of a stylized 'E' icon above the word 'DevNAVI'. On the left side, there is a sidebar with five items labeled 'Problem 1' through 'Problem 5'. To the right of the sidebar, there is a large question card with a circular icon containing a brain-like symbol. The text on the card reads: '질문 1. 가장 자신있는 프로그래밍 언어는 무엇이고, 해당 언어를 선택한 이유는 무엇인가요?'. Below this is a response area with a person icon and a text input placeholder '답을 입력하세요'. In the bottom right corner of the main area, there is a timer showing '00:01:28' and a dark blue button labeled '제출'.

CI/CD

- GitHub의 dev branch로 코드가 merge될 경우 자동으로 AWS의 EC2 인스턴스에 front-end와 back-end 배포
- GitHub actions, AWS의 S3, codedeploy 등을 활용
- 현재 nginx 서버와 pm2 서버를 통해 프론트 서버와 백 서버가 동작 중





Cooperation

- Notion
- GitHub



Notion

프로젝트에 필요한 정보와 문서를 정리

Member

| 이름 | Github ID | e-mail | 역할(임시) |
|-----|--------------|----------------------|----------|
| 권민성 | minsung1386 | [REDACTED]ail.com | frontend |
| 서주원 | userjuwon | [REDACTED]@gmail.com | frontend |
| 안서현 | lincret | [REDACTED]ail.com | backend |
| 안정민 | AnJeongMin | [REDACTED]ail.com | backend |
| 이장엽 | milda-regret | [REDACTED]mail.com | backend |
| 한수현 | SuhyunCation | [REDACTED]kku.edu | frontend |

프로젝트 명세

- ❖ Requirements Specification
- ❖ Design Specification

개발 문서

- ❖ 개발 컨벤션
- ❖ 코드 컨벤션
- ❖ To-do List
- ❖ 기술 스택 선택

웹 관련 참고 자료

공통

JS

- 모던 JavaScript 튜토리얼
- MDN JS (JS document)
- JS - 코딩 컨벤션
- JS - 네이밍 컨벤션
- 많이 쓰이는 es6, es11 문법
- TS document (프론트는 JS)

GitHub

- commit 컨벤션
- git-flow (브랜치 전략)

Web

프론트엔드

HTML & CSS

- 자료 찾아보면 많음
- 생활코딩같은 튜토리얼 따라가 보기
- <https://getbootstrap.com/>

React

- 자습서: React 시작하기

기존에는 class문법을 사용한 컴포넌트를 사용했지만 요즘은 대부분 **함수 컴포넌트**를 사용하므로 권장 자료

Webpack (번들러)

백엔드

NodeJS

- node 18.12.0 (LTS 사용 예정)

NestJS

- [NestJS document](#)
- [NestJS 번역](#)
- [NestJS를 사용하는 이유](#)
- [nestJS 아키텍처](#)
- [nestJS 프로젝트 예제](#)

Architecture

- [배에 드 아키텍처](#)

- ESLint (각 front, back 폴더에 적용)

```
{  
  "env": {  
    "browser": true,  
    "es2021": true  
  },  
  "extends": [  
    "eslint:",  
    "plugin:react/recommended",  
    "plugin:@typescript-eslint/recommended",  
    "prettier"  
  ],  
  "overrides": [],  
  "parser": "@typescript-eslint/parser",  
  "parserOptions": {  
    "ecmaVersion": "latest",  
    "sourceType": "module"  
  },  
  "plugins": ["react", "@typescript-eslint"],  
  "rules": {  
    "prefer-arrow-callback": "error",  
    //하단은 프론트엔드 emotionjs용 lint 설정  
    "react/no-unknown-property": ["error", { "ignore": ["css"] }]  
  }  
}
```

- Naming Convention

- Common

- variables: **camelCase**
 - class: **PascalCase**
 - api URL: **kebab-case**
 - socket event (사용할지는?): **snake_case**

- BE naming convention

- nest file: **do-main.feature.ts**
 - TS type: interface로 통일, **PascalCase** + **Type**

- FE naming convention

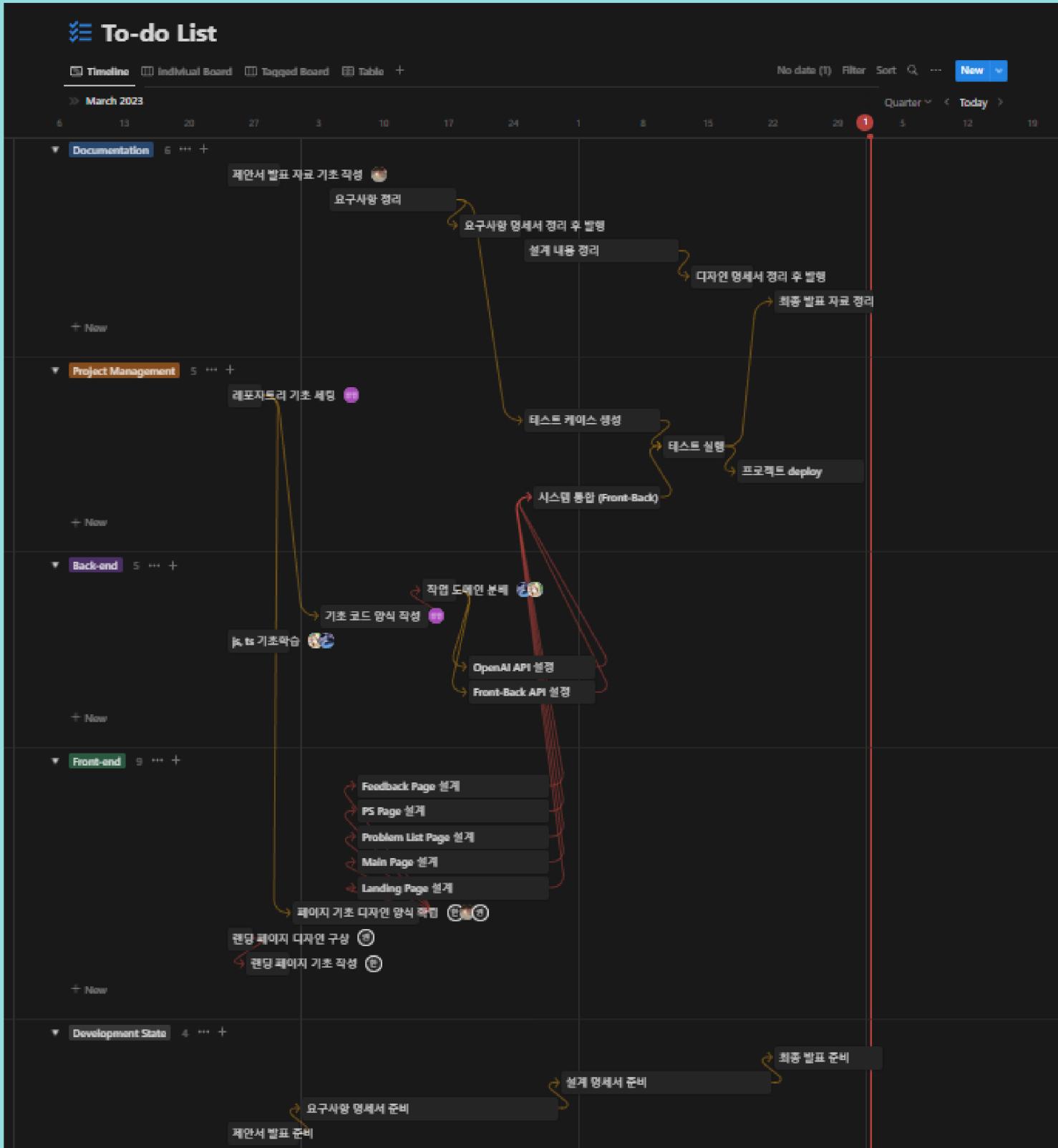
- react component: **PascalCase**

- Prettier (각 front, back 폴더에 적용)

```
{  
  "singleQuote": true,  
  "parser": "typescript",  
  "semi": true,  
  "useTabs": true,  
  "printWidth": 100,  
  "tabWidth": 4  
}
```

일관성 있는 개발 스타일을 위해
Naming Convention 설정하고,
commitizen, prettier, esLint 등을 활용

To-do List를 통해 작업 순서와 일정, 담당자 등을 정리



GitHub

효율적인 프로젝트 개발을 위해
Issue, Pull Request, Review 등의 기능 적극 활용

Git-Flow 전략 활용

브랜치를 최소 컴포넌트 단위로 분리하여
작업 영역을 최대한 분리

Pull Request에 대한 review

상호 피드백에 따른 코드 수정 및 보완

server/src/problem/repository/problem.repository.ts

```
14 +         const createdProblem = new this.problemModel(problem);
15 +         return await createdProblem.save();
16 +     } catch (error) {
17 +         throw new Error(error);
```

AnJeongMin last week

여기서 Error가 node의 Error를 쓴 것 같은데, 그 Error가 아니라 mongoose의 Error를 import해서 써야 mongoose exception filter에 catch 됩니다. `import { Error, Model } from 'mongoose';`로 추가해주세요

milda-regret 3 days ago

수정했습니다!

Any Question?

