

파일 암호화 관리 서버 및 프로그램 개발

프론티어

진예희, 박연희

Contents 목차

- 01 | 프로젝트 개요
- 02 | 프로젝트 팀 구성 및 역할
- 03 | 프로젝트 수행 절차 및 방법
- 04 | 프로젝트 수행 결과
- 05 | 자체평가 의견

01. 프로젝트 개요

- 프로젝트 주제 및 기획의도



파일 암호화 관리 서버 및 프로그램 개발



주제

PC에 설치되어 동작하는 파일 암호화 프로그램 개발 및 프로그램
동작 여부와 작업 로그 등을 관리할 수 있는 서버를 구축 /
파일 복호화 후 재암호화 기능 수행 프로그램 개발

01. 프로젝트 개요

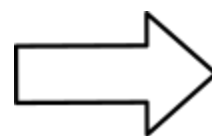
- 프로젝트 주제 및 기획의도



기획의도

파일을 암호화 후 작업 이력을 서버에서 통합 관리할 수 있는 보안 시스템 구축

현대 사회에서 데이터의 유출이 큰 피해로 이어질 수 있으며, 기업 및 개인 모두 **데이터 보안의 중요성이 증가**



사용자 PC에 설치되어 작동하는 프로그램을 개발함과 동시에
프로그램의 동작 여부와 관리자 활동 로그를 통합 관리하는
서버를 구축하여 보안 강화

사용자가 특정 파일을 복호화 후 재암호화 함으로써
정보 유출 가능성을 최소화

01. 프로젝트 개요

- 프로젝트 기술서

1 프로젝트 구현 내용 및 컨셉

최고 관리자와 중간 관리자의 서버 조회
및 관리 기능 구축

파일 암호화 프로그램 작업 시 자동으로
로그를 생성 및 관리

기업 내 보안 강화를 위한 통합 관리
서버 구현

사용자가 파일 복호화 후 재암호화 기능 구현

2 훈련 내용 관련 기술

사용자 인증, 권한 설정(최고/중간) 등
보안 기능 구현

로그 저장 및 서버 전송 기능 구현
(백엔드 및 서버 구축)

데이터베이스 설계 및 구현

파일 재암호화 시에 원본 및 백업 파일 삭제

01. 프로젝트 개요

- 개발 환경

사용 언어

화면 구현 : HTML, CSS, JS

사용자 접근 관리 : PHP

Agent / Atool 프로그램 : C#



디자인

와이어 프레임, 프로토타입 :

Figma



툴

구현 : VSCODE

DB : MariaDB



01. 프로젝트 개요

- 프로젝트 구조

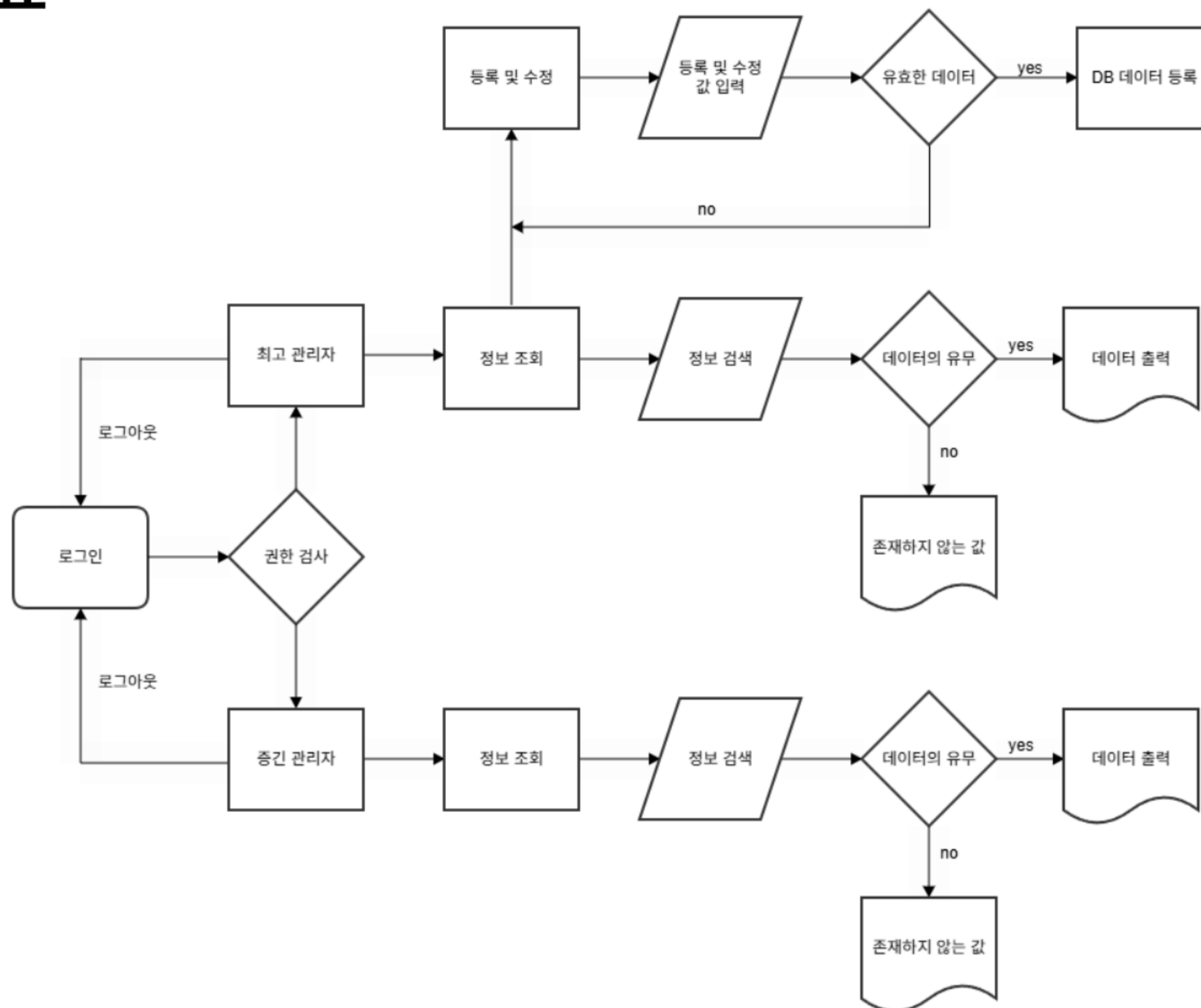
네비게이션 설계



01. 프로젝트 개요

- 프로젝트 구조

워크플로우



01. 프로젝트 개요

- 프로젝트 구조

ERD



01. 프로젝트 개요

- 활용방안 및 기대효과

문서 암호화 프로그램 관리 서버 및 프로그램 개발



문서 유출 방지

중요한 문서가 외부로 유출되는 것을 방지할 수 있습니다.

일관된 정책 적용

관리자가 사용자의 임의 변경을 막고, 기업 전체의 보안 기준을 통일할 수 있습니다.

접근 제어 및 권한 관리

어떤 사용자가 어떤 문서를 열었는지, 언제 접근했는지 등 활동 로그를 중앙 서버에서 기록하고 확인할 수 있습니다.

02. 프로젝트 팀 구성 및 역할

프론티어
The FRONTIER

"새로운 도전을 받아들이고, 한계를 돌파하자"

팀장 진예희

- 백엔드 담당
- DB 구축
- Atool 프로그램 구현

팀원 권소현

- 프론트엔드 담당
- 화면의 UI 디자인

팀원 박연희

- 프론트엔드 담당
- 화면의 UI 디자인 및 구현
- Agent 프로그램 구현

팀원 전진하

- 프론트엔드 담당
- UI 디자인 및 개발 보조

03. 프로젝트 수행 절차 및 방법

수행	1주차					2주차					3주차					4주차					5주차				
프로젝트 기획																									
와이어프레임 설계																									
스타일 가이드 지정																									
프로토타입 구현																									
화면 구현 (프론트엔드)																									
데이터베이스 설계																									
기능 구현 (백엔드)																									
Agent 프로그램 설계 및 구현																									
Atool 프로그램 구현																									

04. 프로젝트 수행 결과

- UI/UX 설계 (관리자 페이지 와이어프레임)

관리자 로그인 페이지

Welcome to Our Service
안전한 서비스 이용을 위해 로그인 해주세요!

Logo or GIF

Login

Id

pwd

☐ 로그인 상태 유지

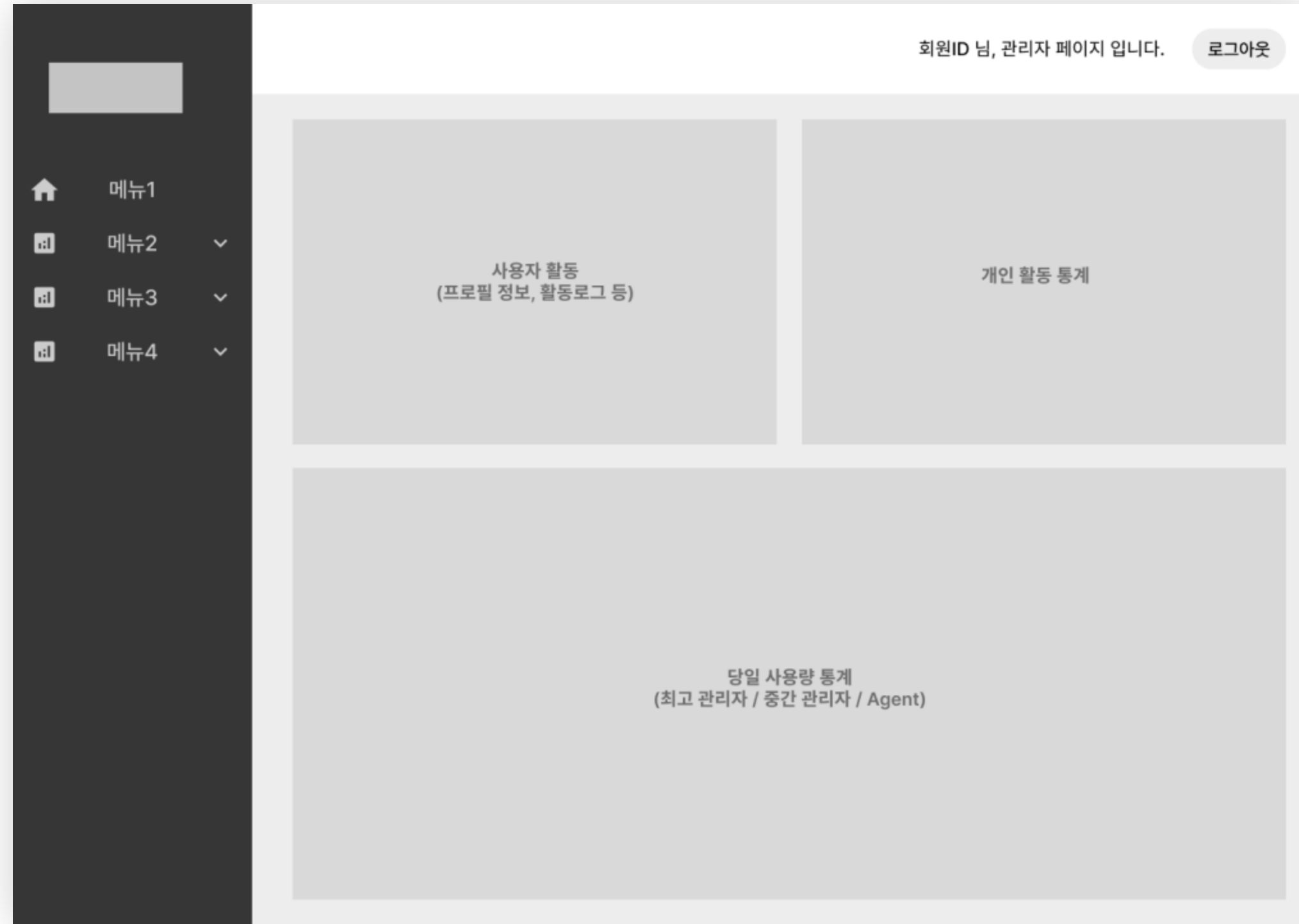
로그인

아직 계정이 없으신가요? [회원가입](#)

04. 프로젝트 수행 결과

- UI/UX 설계 (관리자 페이지 와이어프레임)

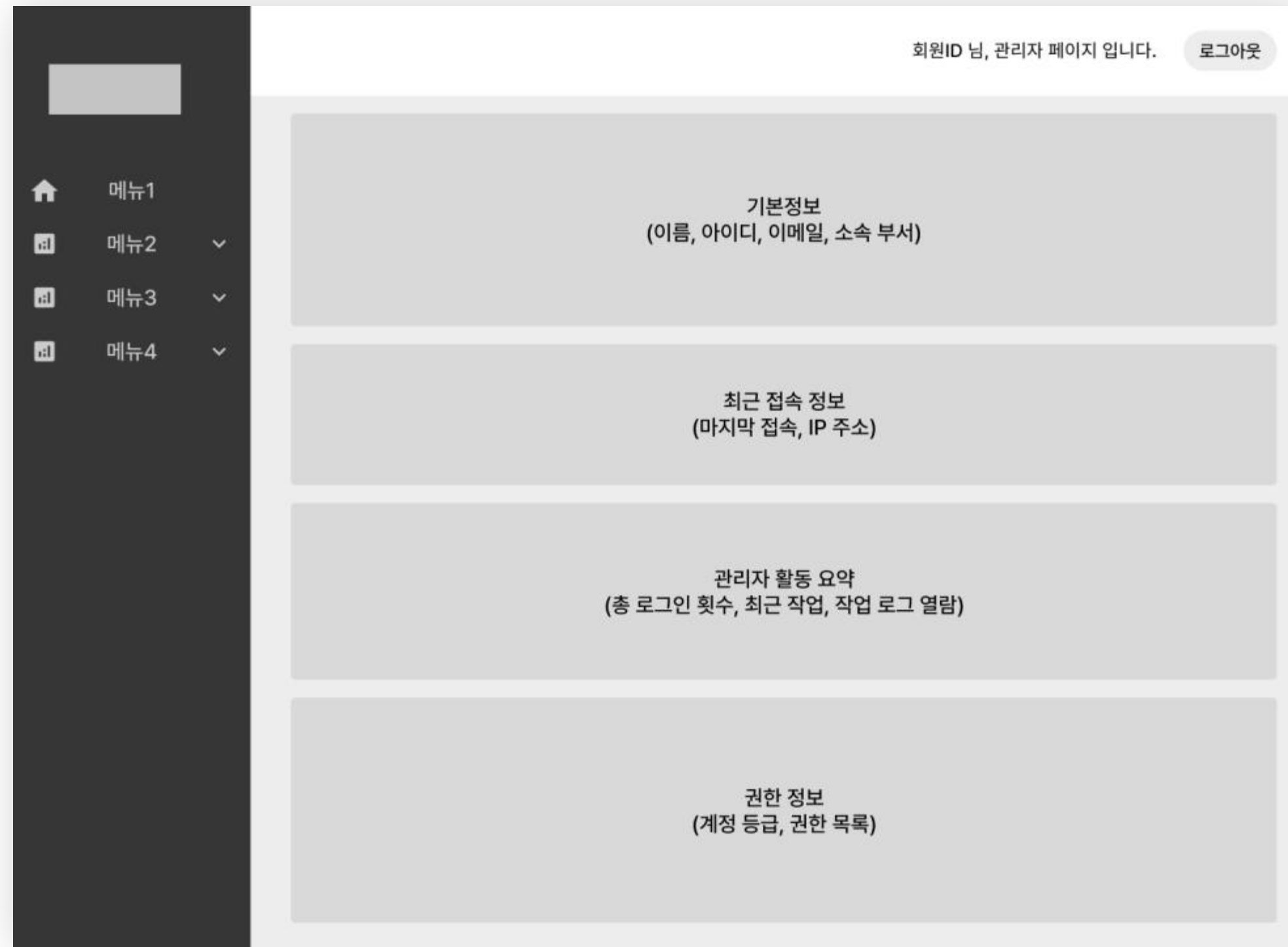
관리자 대시보드



04. 프로젝트 수행 결과

- UI/UX 설계 (관리자 페이지 와이어프레임)

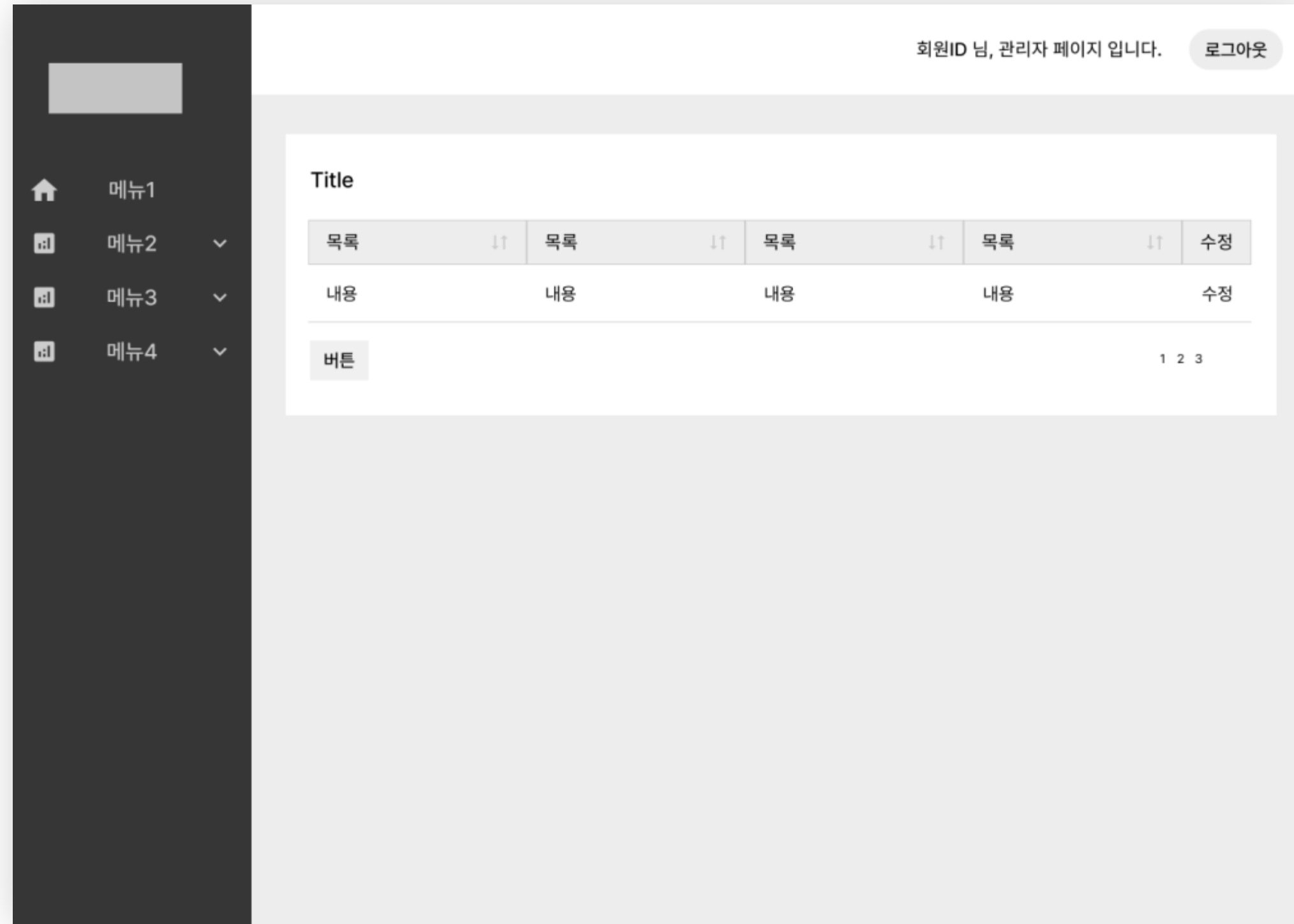
관리자 마이페이지



04. 프로젝트 수행 결과

- UI/UX 설계 (관리자 페이지 와이어프레임)

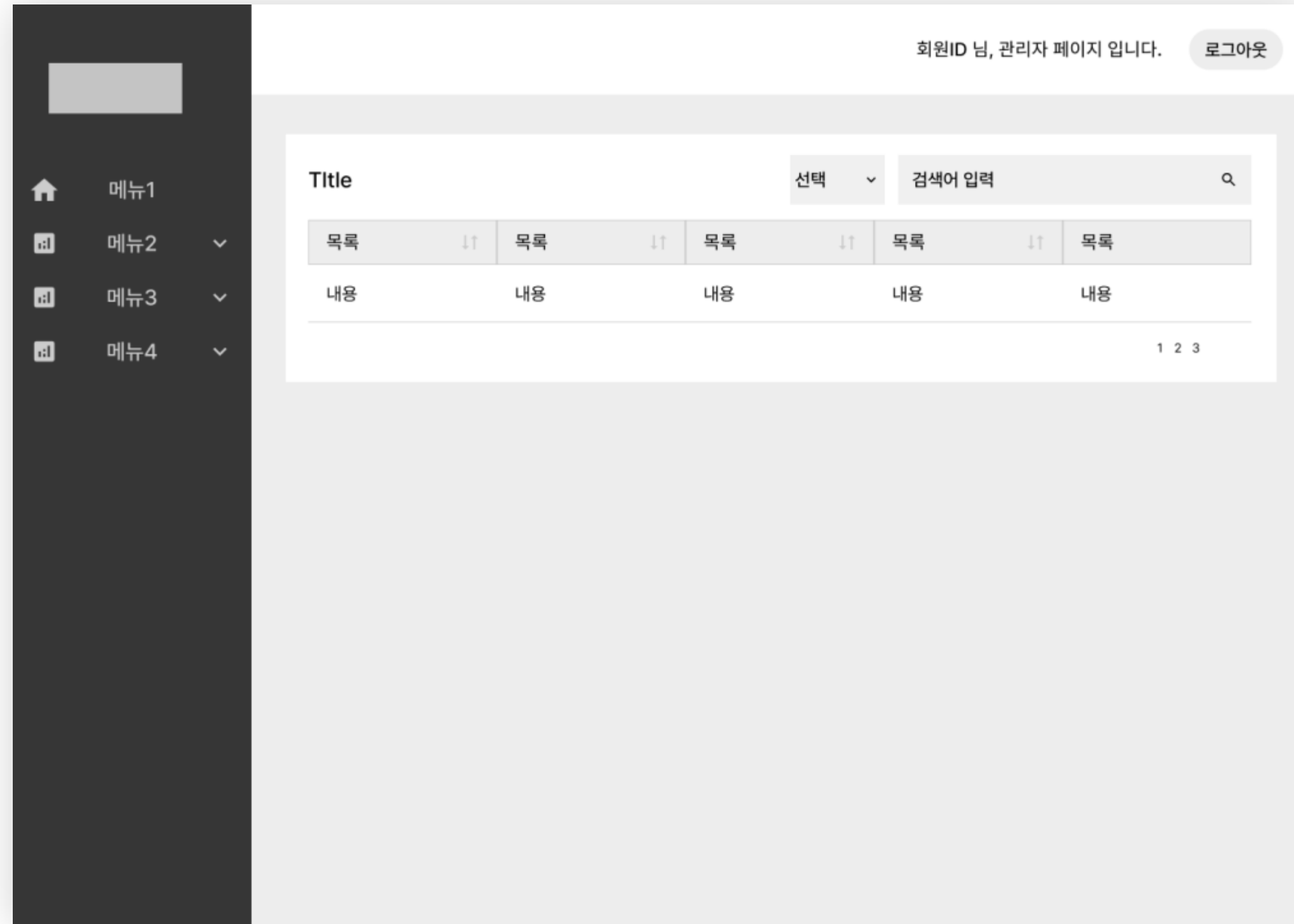
조회 페이지



04. 프로젝트 수행 결과

- UI/UX 설계 (관리자 페이지 와이어프레임)

조회 & 검색 페이지



04. 프로젝트 수행 결과

- UI/UX 설계 (관리자 페이지 와이어프레임)

등록 & 수정 페이지

The wireframe illustrates an administrator interface. On the left is a dark sidebar with a home icon and four menu items labeled '메뉴1' through '메뉴4', each with a dropdown arrow. The main content area has a top header with the text '회원ID 님, 관리자 페이지 입니다.' and a '로그아웃' button. Below this is a 'Title' label. The form contains several input fields: '수정 X 내용', '날짜 내용' (with a calendar icon), '시간 내용' (with a clock icon), '드롭다운 선택 내용' (with a dropdown arrow), and '텍스트 내용'. At the bottom left is a checkbox labeled '체크박스 내용'. At the bottom right are two '버튼' (button) labels.

04. 프로젝트 수행 결과

- UI/UX 설계 (Agent 프로그램 와이어프레임)

회원가입

로고

Hello, I'm imon!

부서명

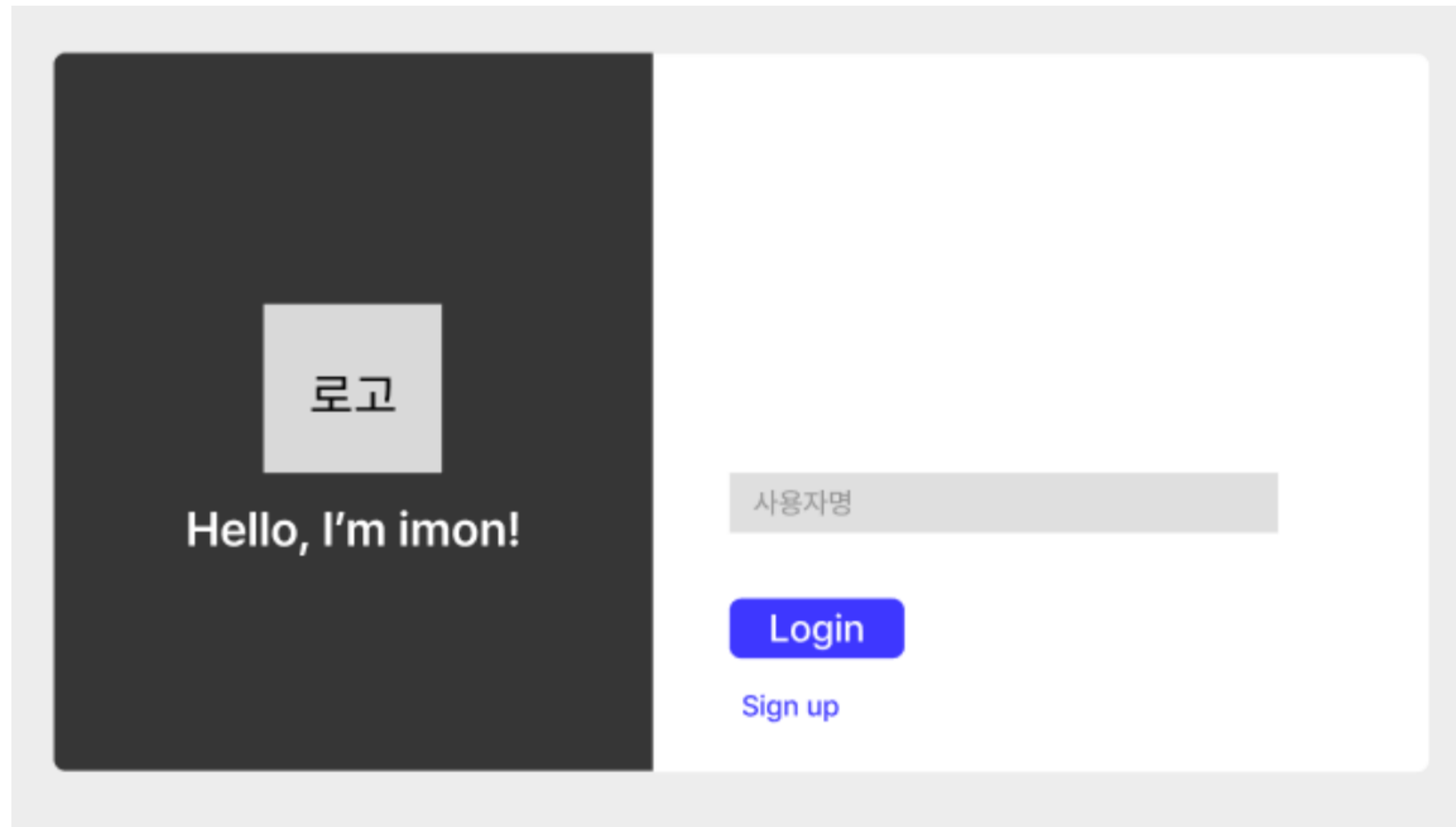
사용자명

Sign up

04. 프로젝트 수행 결과

- UI/UX 설계 (Agent 프로그램 와이어프레임)

로그인



04. 프로젝트 수행 결과

- UI/UX 설계 (Agent 프로그램 와이어프레임)

반출 요청 페이지

The wireframe shows a web interface for the 'imon' system. It features a dark header with the 'imon' logo. A left sidebar contains three menu items: '반출 요청' (Check-out Request) with a document icon, '반출 확인' (Check-out Confirmation) with a checkmark icon, and '정보 전송' (Information Transfer) with an information icon. The main content area is titled 'name' and contains a form with a text input field, a '가져오기' (Load) button, a large '사유' (Reason) text area, and a '요청' (Request) button at the bottom.


04. 프로젝트 수행 결과

- UI/UX 설계 (Agent 프로그램 와이어프레임)


반출 확인 페이지

imon


name



반출 요청



반출 확인



정보 전송

검색

외부 반출 대상	처리

04. 프로젝트 수행 결과

- UI/UX 설계 (Agent 프로그램 와이어프레임)

정보 요청 페이지

imon

name



반출 요청



반출 확인



정보 전송

암호화 대상 확장자

예외 폴더

04. 프로젝트 수행 결과

- UI/UX 디자인 (스타일가이드)

COLOR 색상

Main Color



#363636



#EDEDED



#58C6F4

Sub Color



#939393



#CCCCCC

FONT 글꼴

프리텐다드 Regular - 25px

프리텐다드 Bold - 20px

프리텐다드 Regular - 17px

프리텐다드 Regular - 15px

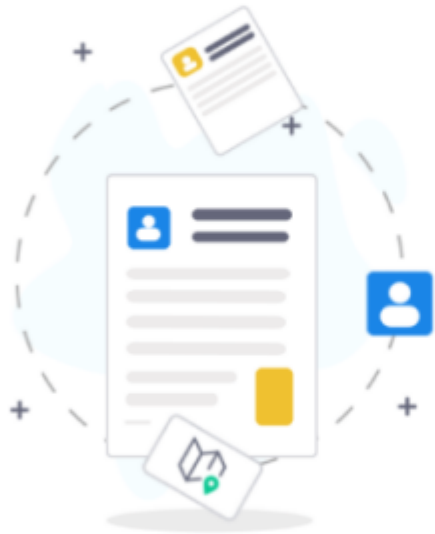
프리텐다드 Regular - 12px

04. 프로젝트 수행 결과

- UI/UX 디자인 (관리자 페이지 프로토타입)

로그인 페이지

Welcome to Our Service
안전한 서비스 이용을 위해 로그인해 주세요



Login

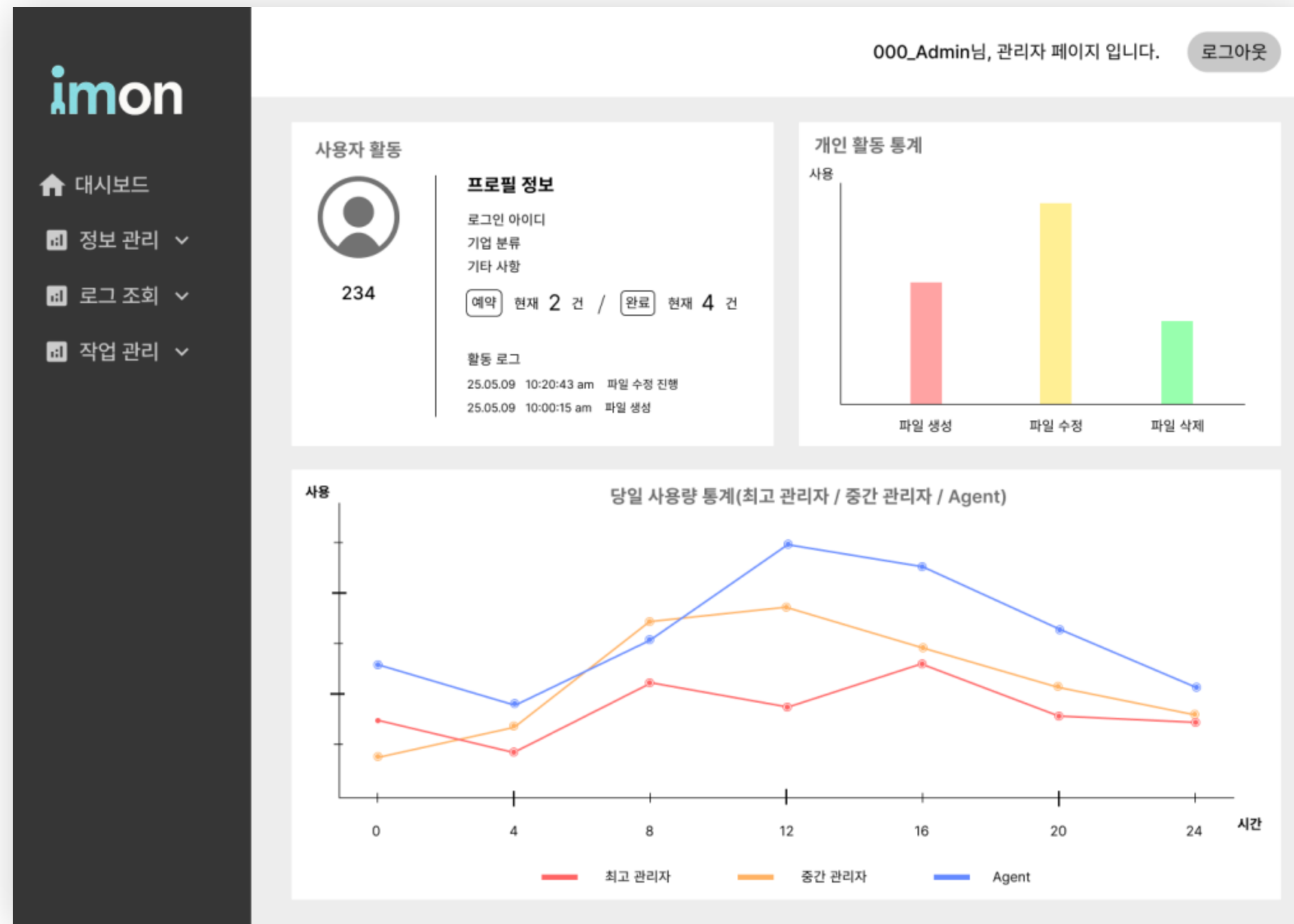
☐ 로그인 상태 유지

[아직 계정이 없으신가요? 회원가입](#)

04. 프로젝트 수행 결과

- UI/UX 디자인 (관리자 페이지 프로토타입)

관리자 대시보드



04. 프로젝트 수행 결과

- UI/UX 디자인 (관리자 페이지 프로토타입)

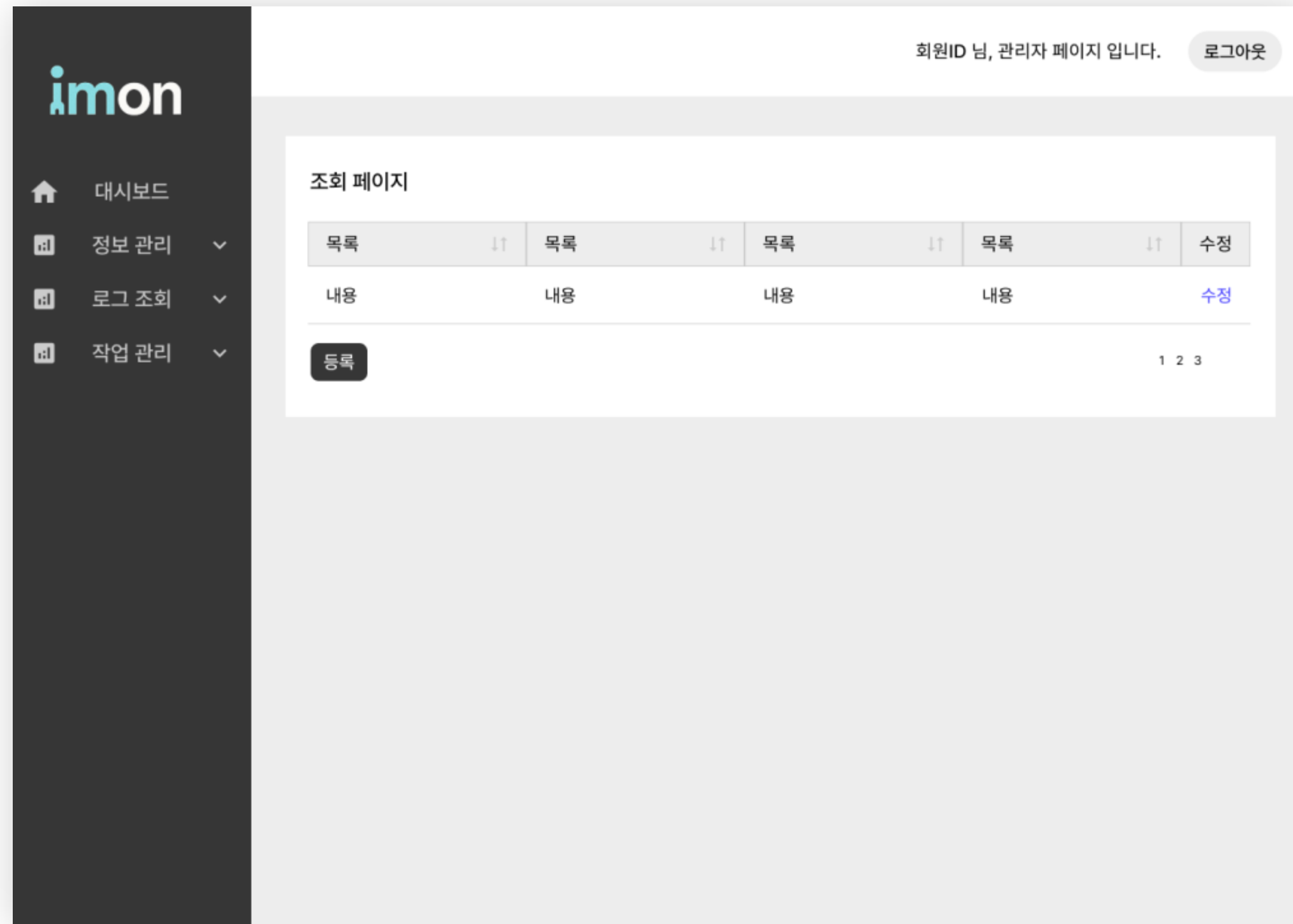
관리자 마이페이지



04. 프로젝트 수행 결과

- UI/UX 디자인 (관리자 페이지 프로토타입)

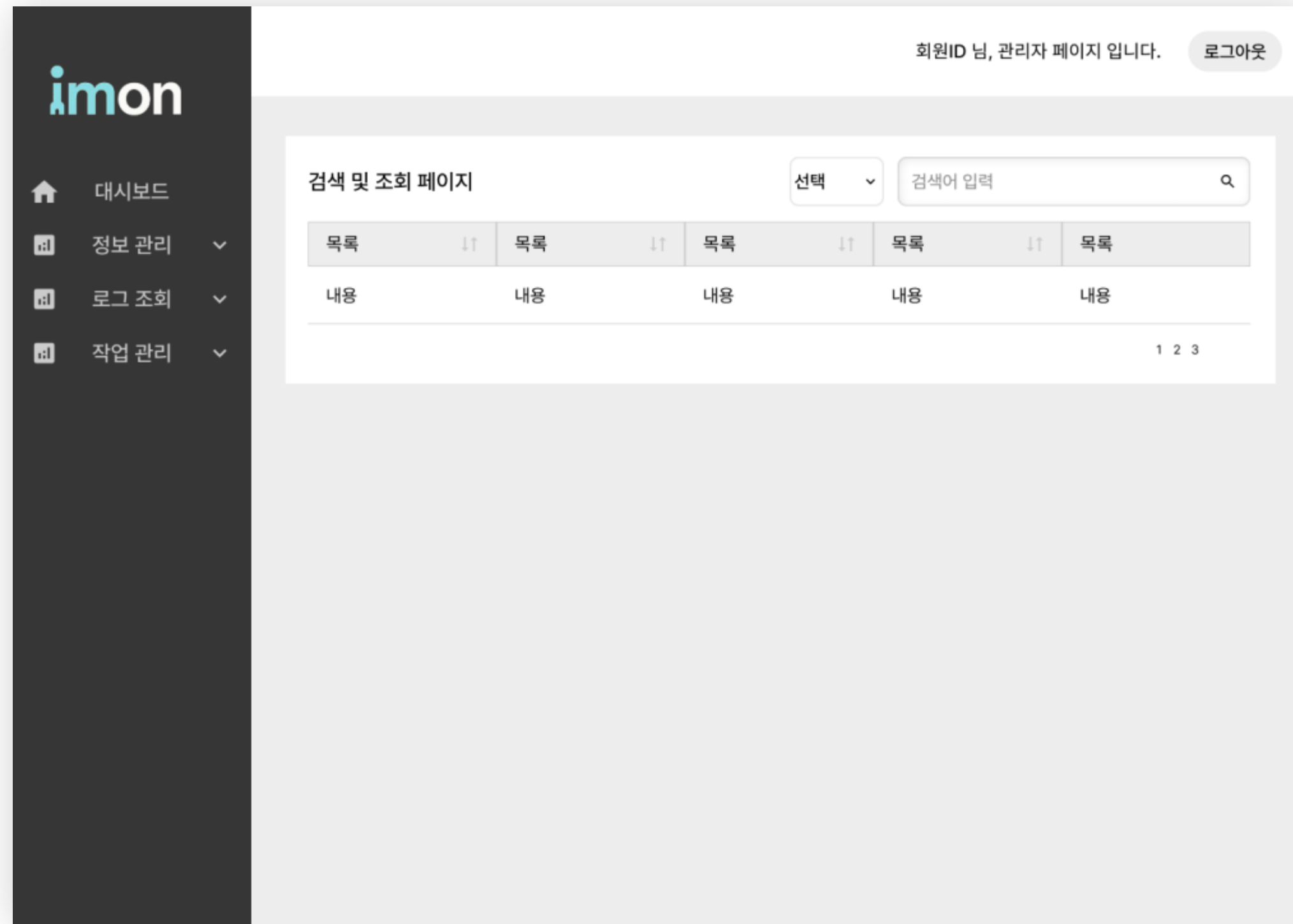
조회 페이지



04. 프로젝트 수행 결과

- UI/UX 디자인 (관리자 페이지 프로토타입)

조회 & 검색 페이지



04. 프로젝트 수행 결과

- UI/UX 디자인 (관리자 페이지 프로토타입)

I 등록 & 수정 페이지

The image shows a web application prototype for the 'imon' system. It features a dark sidebar on the left with the 'imon' logo and navigation links: '대시보드' (Dashboard), '정보 관리' (Information Management), '로그 조회' (Log Search), and '작업 관리' (Task Management). The main content area is titled '등록 및 수정 페이지' (Registration and Modification Page). It contains several form fields: '수정 X 내용' (Modification X Content), '날짜 내용' (Date Content) with a calendar icon, '시간 내용' (Time Content) with a clock icon, '드롭다운 선택 내용' (Dropdown Selection Content), '텍스트 내용' (Text Content), and a checkbox for '체크박스 내용' (Checkbox Content). At the bottom right, there are two buttons: '취소' (Cancel) and '등록' (Register).

imon

회원ID 님, 관리자 페이지 입니다. [로그아웃](#)

등록 및 수정 페이지

수정 X 내용

날짜 내용

시간 내용

드롭다운 선택 내용

텍스트 내용

☐ 체크박스 내용

[취소](#) [등록](#)

04. 프로젝트 수행 결과

- UI/UX 설계 (Agent 프로그램 프로토타입)

회원가입

Imon

Hello, I'm imon!

부서명

사용자명


등록

로그인

04. 프로젝트 수행 결과

- UI/UX 설계 (Agent 프로그램 프로토타입)

로그인



The image shows a login screen prototype. On the left, a dark gray vertical panel contains a blue circular logo with a white shield and checkmark, labeled 'Imon' below it. Below the logo, the text 'Hello, I'm imon!' is displayed. On the right, a white background contains a light gray input field labeled '사용자명' (Username). Below the input field is a blue button with the text '로그인' (Login). At the bottom right, the text '등록' (Register) is displayed in blue.

04. 프로젝트 수행 결과

- UI/UX 설계 (Agent 프로그램 프로토타입)

반출 요청 페이지

The image shows a UI/UX design for a 'Return Request' page. The page has a dark header bar with the 'Imon' logo (a blue circle with a white checkmark) on the left and the text '체크 요청' and '로그아웃' on the right. Below the header, the page is divided into two main sections. The left section, titled '사용자이름님', contains three vertically stacked buttons: '반출 요청' (Return Request) with a document icon, '반출 확인' (Return Confirmation) with a checkmark icon, and '정보 요청' (Information Request) with an 'i' icon. The right section contains a large text input area with the placeholder text '파일을 드래그하거나 가져와주세요' (Drag or drop files here) and a '가져오기' (Upload) button. Below this input area is a '요청' (Request) button. The entire page is enclosed in a light gray border.

04. 프로젝트 수행 결과

- UI/UX 설계 (Agent 프로그램 프로토타입)

반출 확인 페이지



체크 요청 로그아웃

사용자이름님



반출 요청



반출 확인



정보 요청

요청ID

검색

외부 반출 대상	처리

04. 프로젝트 수행 결과

- UI/UX 설계 (Agent 프로그램 프로토타입)

정보 요청 페이지



체크 요청 로그아웃

사용자이름님



반출 요청



반출 확인



정보 요청

암호화 대상 확장자

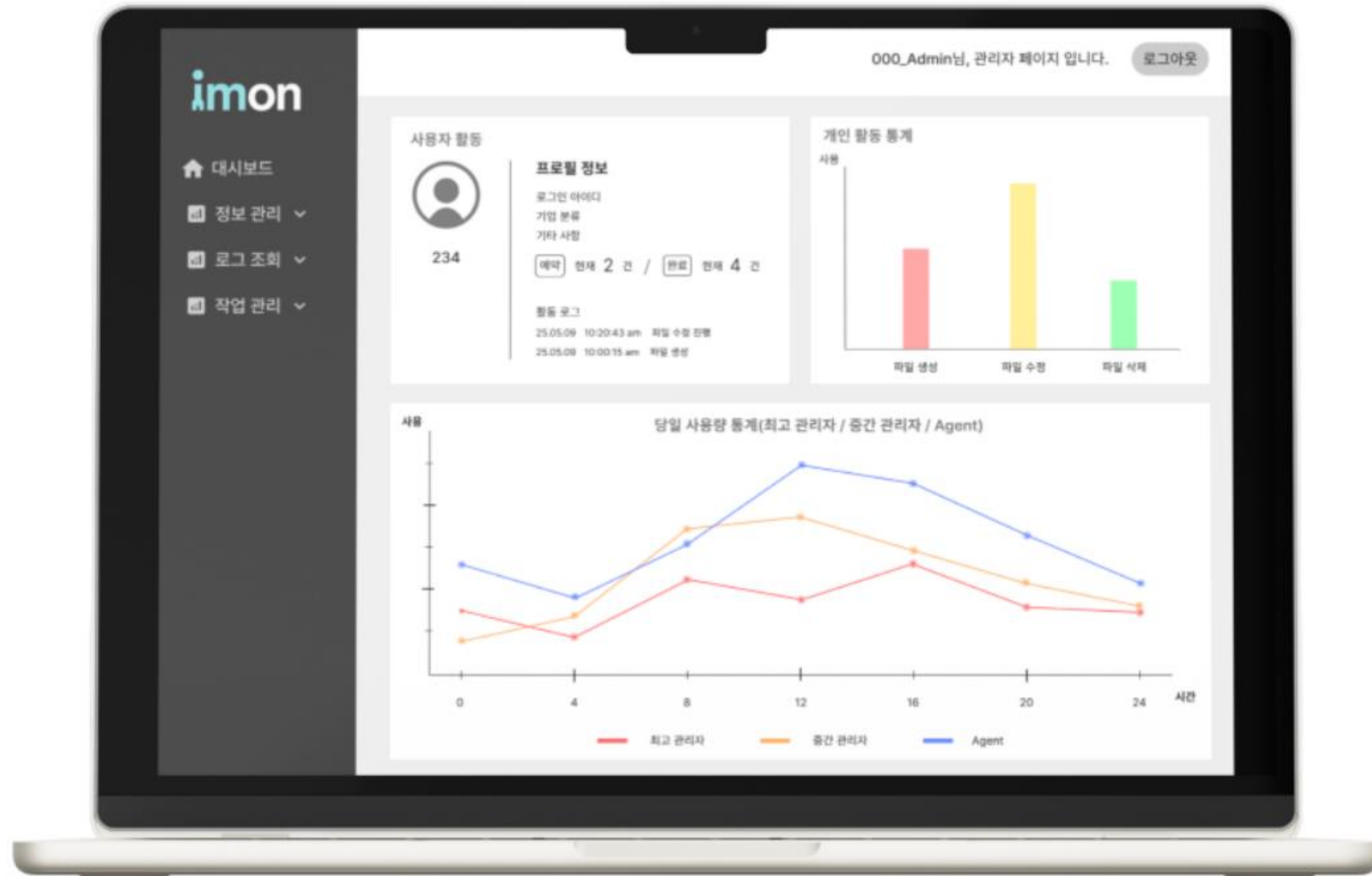
.txt|.hwp|.doc

예외 폴더

C:\Windows\C:\Doc

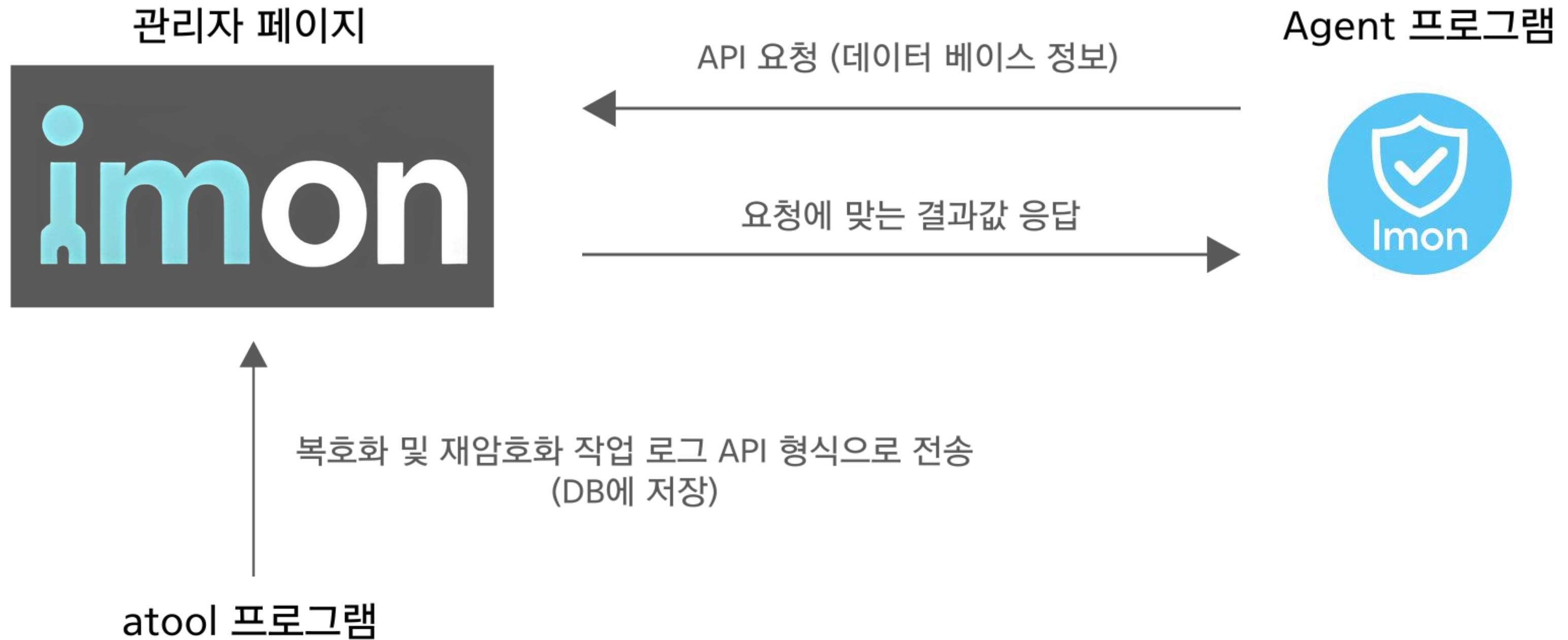
04. 프로젝트 수행 결과

- UI/UX 디자인 (목업)



04. 프로젝트 수행 결과

- 기술 구현



04. 프로젝트 수행 결과

- 기술 구현 (Api 응답)

Api 주소	응답문
check.html (체크 주기 Api)	OK agent 체크주기
get_departments.html (부서명 드롭다운 Api)	OK 부서명들
getconfig.html (부서 설정 정보 리턴 Api)	OK 암호화 대상 확장자 예외폴더
login.html (agent 로그인 Api)	OK agent 체크주기 agent이름
logout.html (agent 로그아웃 Api)	OK
register.html (agent 등록 Api)	OK agentId agent 체크주기
report.html (agent 암호화 작업 로그 전송 Api)	OK
request.html (외부 반출 요청 Api)	OK 요청 ID
requestcheck.html (외부 반출 확인 Api)	OK 반출대상 반출결과

04. 프로젝트 수행 결과

- 기술 구현 (Agent 프로그램)

```
private async void RegisterApi()  
{  
    string userName = Textbox2.Text;           사용자의 입력값 받아오기  
    string departName = comboBox1.Text;  
    string val = $"{hostName}|{ipAddress}|{userName}|{departName}";  
    string url = $"{baseApiUrl}register.html?val=" + Uri.EscapeDataString(val);  Api 루트 지정  
  
    try  
    {  
        HttpResponseMessage response = await client.GetAsync(url);  Api 요청  
  
        if (response.IsSuccessStatusCode)  
        {  
            string result = await response.Content.ReadAsStringAsync();  Api 응답  
  
            if (result.StartsWith("OKI"))  
            {  
                string[] parts = result.Split('|');  응답을 배열로 분리  
                string agentId = parts[1];  
                string interval = parts[2];  
                UserSession.Username = userName;  
                MessageBox.Show($"Agent 등록 성공\nID: {agentId}\n주기: {interval}");  
  
                string filePath = Path.Combine(Environment.GetFolderPath(Environment.SpecialFolder.ApplicationData), "AgentID.id");  
  
                File.WriteAllText(filePath, agentId);  agentID이 담긴 파일을 사용자의 컴퓨터에 생성  
  
                LoginApi();  
            }  
        }  
    }  
}
```

04. 프로젝트 수행 결과

- 기술 구현 (Atool 프로그램)

```
static void DecryptFile(string encryptedFilePath, string decryptedFilePath)
```

AES로 암호화된 파일을 다시 복호화

```
{  
    string extension = Path.GetExtension(decryptedFilePath).ToLower();
```

```
    using (FileStream input = new FileStream(encryptedFilePath, FileMode.Open, FileAccess.Read))
```

```
    using (FileStream output = new FileStream(decryptedFilePath, FileMode.Create, FileAccess.Write))
```

암호화 된 입력 파일을 FileStream으로 열기 / 원본파일 생성

```
{  
    try
```

```
{
```

```
        using (Aes aes = Aes.Create()) Aes 객체 생성
```

```
{
```

```
            aes.Key = aesKey;
```

지정된 키(aes.Key)와 초기 벡터(aes.IV) 설정

```
            aes.IV = aesIV;
```

파일을 암호화한 프로그램과 같아야 한다.

```
            using (CryptoStream cryptoStream = new CryptoStream(input, aes.CreateDecryptor(), CryptoStreamMode.Read))
```

```
{
```

```
                if (extension == ".txt")
```

```
{
```

```
                    using (var reader = new StreamReader(cryptoStream, Encoding.UTF8))
```

```
                    using (var writer = new StreamWriter(output, Encoding.UTF8))
```

```
{
```

```
                        writer.Write(reader.ReadToEnd());
```

```
}
```

```
}
```

```
            else
```

```
{
```

```
                cryptoStream.CopyTo(output);
```

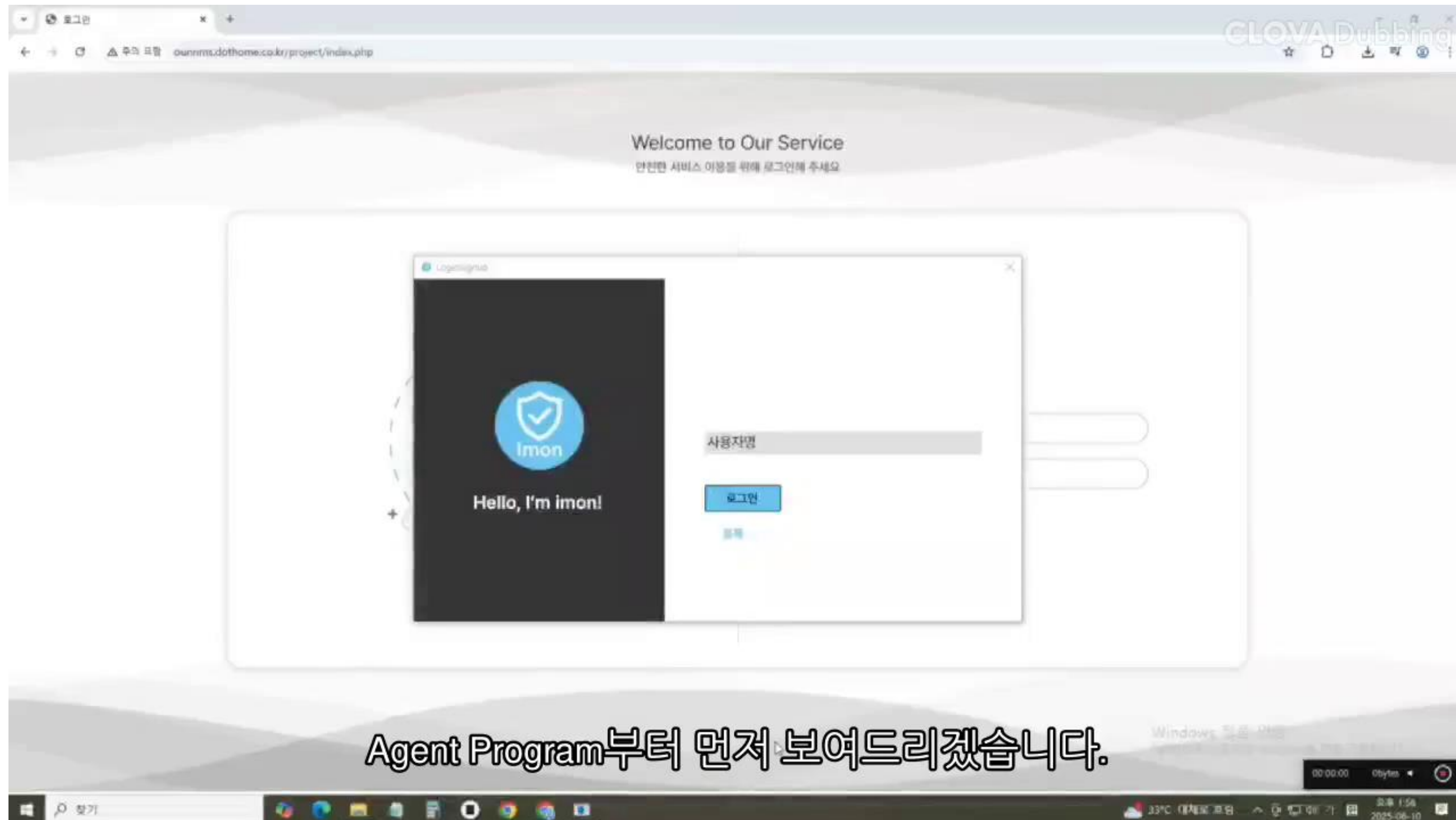
```
}
```

```
}
```

```
}
```


04. 프로젝트 수행 결과

- 시연 동영상



05. 자체평가 및 의견

진예희

완벽하진 않지만 계획했던 부분은 다 구현하였습니다. 처음 접하는 프로그램으로 개발을 진행해서 오류 처리가 아쉽다고 생각하였습니다. 학교에서 진행하던 프로젝트와는 다르게 요구사항을 직접 분석하며 진행해야 하는 프로젝트라 새로움을 느꼈습니다. 평범한 과제로는 경험하지 못할 기술들을 이번 기회를 통해 직접 구현할 수 있어서 즐거웠습니다.

박연희

이번에 일정이 계속해서 밀려서 보충할 부분을 완전히 하지 못하였습니다. 그래서 일정을 짜도 일정대로 움직여지지 않기에 미리미리 해두는 것이 좋다는 생각을 하게 되었습니다. 그리고 처음으로 회사와 같이 진행하는 큰 프로젝트라는 느낌에 부담도 컸지만 결과적으로 계획한 부분을 해낸 것에 대해 뿌듯함을 느끼기도 했습니다.

Q & A

감사합니다.