

# 로그인 페이지

**Welcome!**

coFlo

맞춤형 코드 분석을 통해 코드 리뷰와 개발을 효과적으로 제공하는 플랫폼



**오늘도 coFlo 할까요?**

플로팅과 함께 내 프로젝트 코드를 리뷰하세요.



카카오로 시작하기



구글로 시작하기

소셜 계정을 통해 바로 이용이 가능하며  
첫 로그인시 이용약관 및 개인정보처리방침 동의로 간주됩니다.

소셜 로그인  
(카카오, 구글)

# 회원가입 페이지



가져올 Git 서비스 URL을 선택해주세요.

lab.ssafy.com

사용자 인증 토큰을 입력해주세요.

검증하기

.....|

검증하기 버튼을 눌러 토큰을 검증해주세요.

● 사용자 인증 토큰 안내

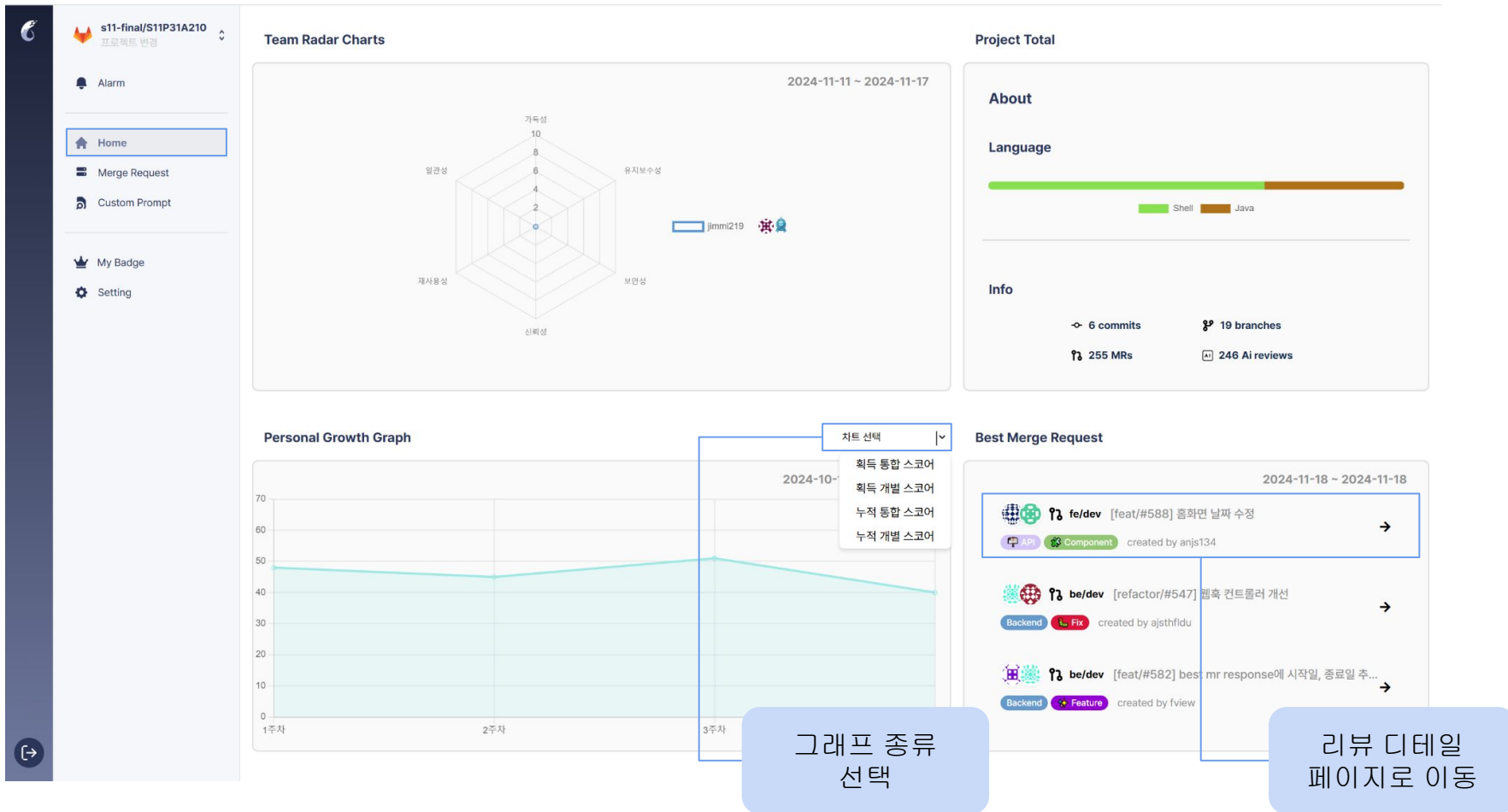
가입하기

연결할 깃랩  
도메인 입력

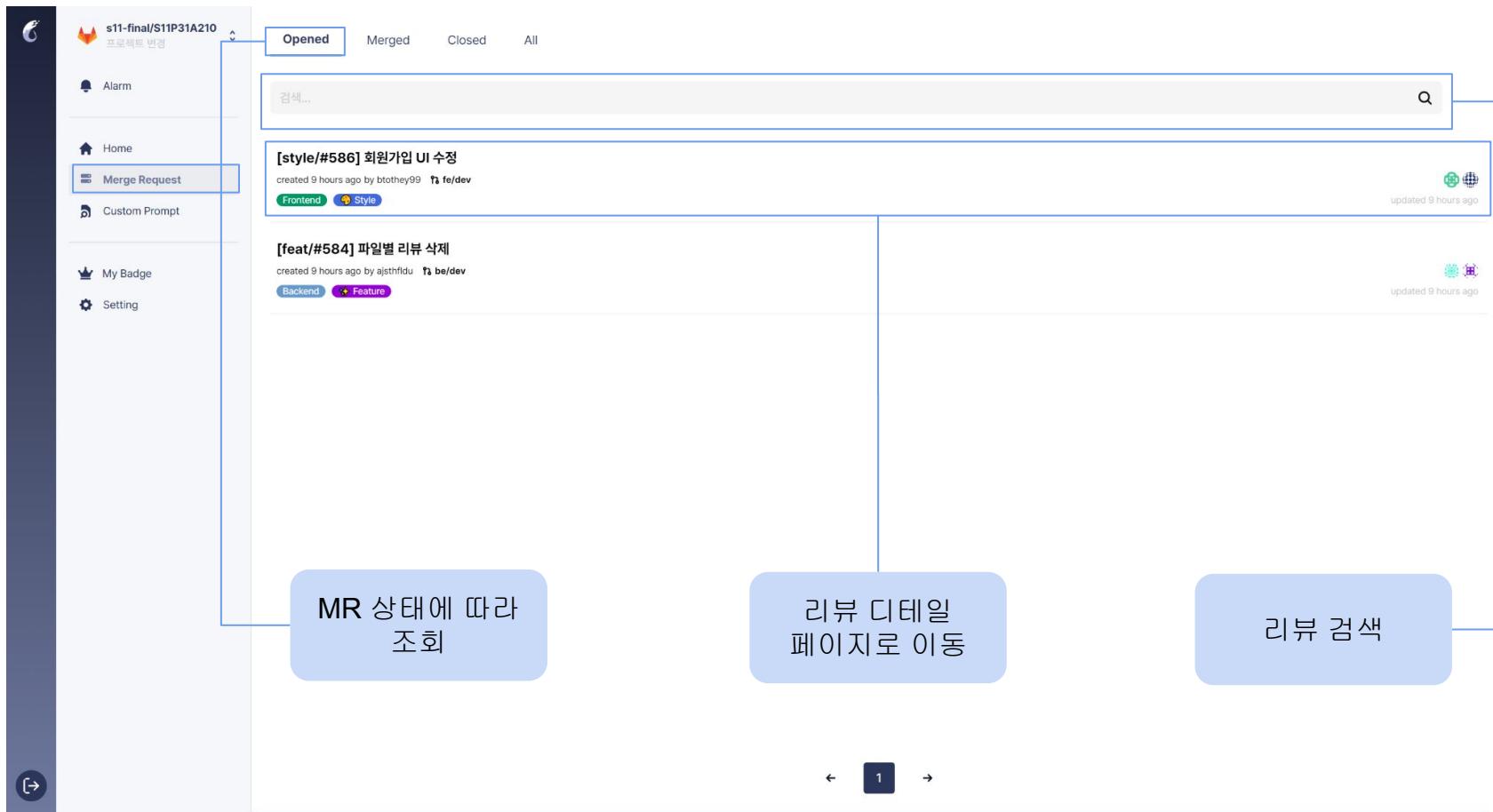
연결할 깃랩  
사용자 토큰  
입력

회원가입


# 메인 페이지



# Merge Request 리스트 조회 페이지



# 리뷰 디테일 조회 페이지

 s11-final/S11P31A210  
프로젝트 변경

Alarm

Home

Merge Request

Custom Prompt

My Badge

Setting

[refactor/#536] Custom Prompt content 길이제한 1000자로 변경 #199

merged 유승현 requested to merge be/S11P31A210-536 코드 리팩토링 to be/dev

플로핑 reviewed 6 days ago

플로핑 left a comment

변경된 코드에 대한 리뷰를 진행하겠습니다.

코드 리뷰

1. 변경사항의 긍정적 측면

- 길이 제한 추가: `content` 필드에 `@Column(length = 1000)` 어노테이션을 추가하여 최대 길이를 명시한 점은 데이터 무결성을 유지하고, 예기치 않은 데이터 입력으로 인한 오류를 방지하는 좋은 접근입니다.

2. 코드 정확성

- `CustomPrompt` 클래스에 `content` 필드의 길이를 제한하는 변경은 전반적으로 문제를 일으키지 않을 것으로 보입니다. 그러나, 해당 속성의 길이 제한을 어길 경우 발생할 수 있는 예외 처리에 대한 고려가 필요합니다. 예를 들어, 이 필드에 대해 `@Size` 어노테이션을 추가하여 입력 Validation을 강화할 수 있습니다.

3. 가독성

- 코드는 명확하게 작성되었으며, 잘 구조화되어 있어 가독성이 좋습니다. `content`의 길이 제한을 지정하는 어노테이션이 추가된 점은 클래스의 다른 필드를 이해하는 데에도 도움이 됩니다.

4. 효율성 및 성능

- 길이 제한이 추가되며 이로 인해 데이터베이스의 저장 공간이 적절히 관리될 수 있습니다. 지나치게 긴 문자열의 입력을 방지함으로써 저장소 효율성이 향상될 것입니다.

5. 확장성

- 해당 필드에 대한 길이 제한은 다른 기능이나 변경사항에 크게 영향을 미치지 않을 것입니다. 그러나 서비스가 발전하면서 길이 제한이 더 필요하거나 변경되어야 할 경우에 대비해 관련 문서나 주석을 추가하는 것이 좋겠습니다.

보러 가기

전체보기

References

CustomPromptService.java

```
@Transactional public void updateCustomPrompt(User user, String content, Long projectId) { CustomPrompt customPrompt = customPromptRepository.findById(projectId).orElseThrow(() -> new BusinessException(CUSTOM_PROMPT_NOT_EXIST)); customPrompt.updateContent(content); promptHistoryRepository.save(new PromptHistory(user.getId(), LocalDateTime.now())); badgeEventService.eventUpdateCustomPrompt(user); }
```

CustomPromptService.java

```
public CustomPromptResponse getCustomPrompt(Long projectId) { CustomPrompt customPrompt = customPromptRepository.findById(projectId); return CustomPromptResponse.builder().customPromptId(customPrompt.getId()).content(customPrompt.getContent().replace("\n", "<br>")).build(); }
```

ProjectService.java

```
private void saveBasicCustomPrompt(Project project) { CustomPrompt customPrompt = new CustomPrompt(project); customPromptRepository.save(customPrompt); }
```

CustomPromptController.java

```
@PostMapping("/{projectId}") @Operation(summary = "연동된 프로젝트의 Custom Prompt 수정") public ApiResponse<Void> updateCustomPrompt(@AuthUser User user, @PathVariable("projectId") Long projectId, @ModelAttribute CustomPromptRequest customPromptRequest) { customPromptService.updateCustomPrompt(user, customPromptRequest.content(), projectId); return ApiResponse.success(); }
```


CustomPromptController.java

```
@GetMapping("/{projectId}") @Operation(summary = "연동된 프로젝트의 Custom Prompt 조회") public ApiResponse<CustomPrompt> getCustomPrompt(@AuthUser User user, @PathVariable("projectId") Long projectId) { CustomPrompt customPrompt = customPromptRepository.findById(projectId).orElseThrow(() -> new BusinessException(CUSTOM_PROMPT_NOT_EXIST)); return ApiResponse.success(customPromptService.getCustomPrompt(projectId)); }
```

깃랩 MR URL로 이동

참고 자료 조회 페이지로 이동

# 참고자료 조회 페이지



s11-final/S11P31A210  
프로젝트 변경

Alarm

Home

Merge Request

Custom Prompt

My Badge

Setting

## References

내용을 추가하거나 수정하여 리뷰를 재생성 할 수 있습니다.  
모든 변경사항은 재생성시 저장됩니다.

CustomPromptService.java

```
@Transactional
public void updateCustomPrompt(User user, String content, Long projectId) {
    CustomPrompt customPrompt =
        customPromptRepository
            .findByProjectId(projectId)
            .orElseThrow(() -> new BusinessException(CUSTOM_PROMPT_NOT_EXIST));
    customPrompt.updateContent(content);
    promptHistoryRepository.save(new PromptHistory(user.getId(), LocalDateTime.now()));
    badgeEventService.eventUpdateCustomPrompt(user);
}
```

CustomPromptService.java

```
public CustomPromptResponse getCustomPrompt(Long projectId) {
    CustomPrompt customPrompt = customPromptRepository.getByProjectId(projectId);

    return CustomPromptResponse.builder()
        .customPromptId(customPrompt.getId())
        .content(customPrompt.getContent().replace("\n", "<br>"))
        .build();
}
```

ProjectService.java

```
private void saveBasicCustomPrompt(Project project) {
    CustomPrompt customPrompt = new CustomPrompt(project);
    customPromptRepository.save(customPrompt);
}
```

CustomPromptController.java

```
@PostMapping("/{projectId}")
@Operation(summary = "연속된 프로젝트의 Custom Prompt 수정")
public ApiResponse<Void> updateCustomPrompt(
    @AuthUser User user,
    @PathVariable("projectId") Long projectId,
```

참고자료 코드  
수정

참고자료 삭제

# 참고자료 조회 페이지



s11-final/S11P31A210  
프로젝트 변경

Alarm

Home

Merge Request

Custom Prompt

My Badge

Setting

```
GitlabAccount gitlabAccount;
Long gitlabProjectId;
ProjectLinkRequest projectLinkRequest) {
    Project project = createAndSaveProject(gitlabAccount, gitlabProjectId, projectLinkRequest);
    saveBasicCustomPrompt(project);
    branchService.addBranches(projectLinkRequest, project);
    projectWebhookService.addGitlabProjectWebhooks(gitlabAccount.getDomain(), project);
    return project;
}
```

BadgeEventService.java

</> JAVA 347 / 3000

```
1 public void eventUpdateCustomPrompt(User user) {
2     long promptCount = promptHistoryRepository.countByUserId(user.getId());
3     if (promptCount == PROMPT_UPDATE_TARGET_COUNT) {
4         userBadge = UserBadge.of(user, badgeCode);
5         userBadgeRepository.save(userBadge);
6         eventAllBadgeUnlocked(user);
7     }
8 }
```

+

참고자료 추가하기

참고자료 수정


참고자료 추가

리뷰 재생성

리뷰 재생성하기



# 커스텀 프롬프트 페이지



s11-final/S11P31A210  
프로젝트 변경

Alarm


Home

Merge Request

Custom Prompt

My Badge

Setting



## Custom prompt


프로젝트마다 원하는 리뷰 형식으로 요청할 수 있습니다.

현재 저장된 프롬프트

주어진 참고자료를 보고 변경사항을 검토해줘. 놓친 예외케이스가 없는 지 중점적으로 검토해줘!!

원하는 프롬프트를 입력해주세요.

0 / 1000

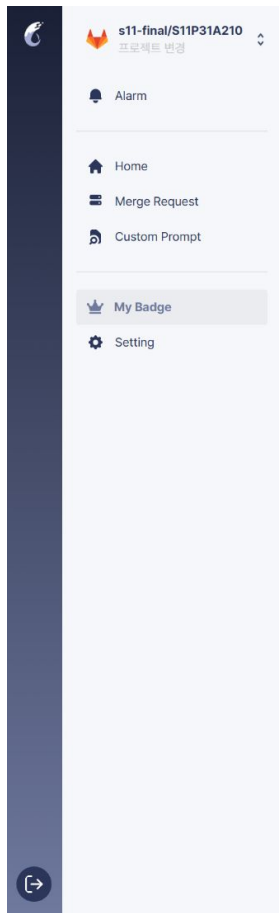


프롬프트 입력

프롬프트 수정

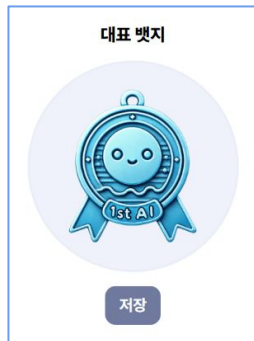


# 배지 페이지



## My Badge

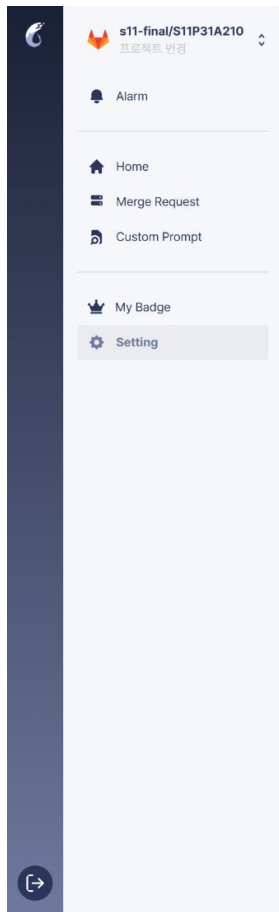
다른 사람에게 보이는 배지를 설정 할 수 있습니다.



대표배지 설정



# 설정 페이지



## Setting

### ● 웹훅 설정

#### 🔗 MATTERMOST

https://meeting.ssafy.com/hooks/zhepiqizm3gtte9cxofzop58sw

설정

테스트 메시지

#### 🗨 DISCORD

URL 입력...

설정

테스트 메시지

웹훅 URL 설정