

# 업데이트된 API 명세서

---

## 1. 인증 (Authentication)

### 1.1 로그인

- **URL:** `/api/token/`
- **Method:** `POST`
- **Content-Type:** `application/json`
- **Request Body:**

```
{
  "username": "string",
  "password": "string"
}
```

- **Response (200):**

```
{
  "access": "string",
  "user": {
    "id": 0,
    "username": "string",
    "email": "string",
    "employee_id": "string",
    "role": "string",
    "rank": "string",
    "department": 0,
    "department_name": "string"
  }
}
```

### 1.2 토큰 갱신

- **URL:** `/api/token/refresh/`
- **Method:** `POST`
- **Request Body:**

```
{
  "refresh": "string"
}
```

- **Response (200):**

```
{
  "access": "string"
}
```

### 1.3 로그아웃

- **URL:** /api/auth/logout/
- **Method:** POST
- **Headers:**

```
Authorization: Bearer {access_token}
```

- **Response (200):**

```
{
  "detail": "로그아웃되었습니다."
}
```

## 2. 사용자 (Users)

### 2.1 현재 사용자 정보 조회

- **URL:** /api/users/me/
- **Method:** GET
- **Response (200):**

```
{
  "id": 0,
  "username": "string",
  "email": "string",
  "employee_id": "string",
  "role": "string",
  "rank": "string",
  "department": 0,
  "department_name": "string",
  "first_name": "string",
  "last_name": "string"
}
```

### 2.2 사용자 목록 조회

- **URL:** /api/users/
- **Method:** GET
- **Query Parameters:**

- **department**: 부서 ID
- **include\_child\_depts**: true/false (기본값: true)
- **rank**: 직급
- **search**: 검색어 (이름, 사번, 이메일)
- **Response (200)**:

```
{
  "count": 0,
  "next": "string(url)",
  "previous": "string(url)",
  "results": [
    {
      "id": 0,
      "username": "string",
      "email": "string",
      "employee_id": "string",
      "role": "string",
      "rank": "string",
      "department": 0,
      "department_name": "string",
      "first_name": "string",
      "last_name": "string"
    }
  ]
}
```

## 2.3 사용자 통계 조회

- **URL**: `/api/users/{id}/tasks_statistics/`
- **Method**: `GET`
- **Response (200)**:

```
{
  "total_tasks": 0,
  "completed_tasks": 0,
  "in_progress_tasks": 0,
  "delayed_tasks": 0,
  "completion_rate": 0.0,
  "tasks_by_priority": {
    "HIGH": 0,
    "MEDIUM": 0,
    "LOW": 0
  }
}
```

## 2.4 사용자 상세 통계 조회

- **URL**: `/api/users/{id}/tasks_statistics_detail/`

- **Method:** GET
- **Response (200):**

```
{
  "priority_distribution": {
    "URGENT": 0,
    "HIGH": 0,
    "MEDIUM": 0,
    "LOW": 0
  },
  "difficulty_distribution": {
    "VERY_HARD": 0,
    "HARD": 0,
    "MEDIUM": 0,
    "EASY": 0
  },
  "avg_completion_time": 0.0,
  "delay_rate": 0.0,
  "monthly_stats": {
    "2024-03": {
      "total": 0,
      "completed": 0,
      "delayed": 0,
      "avg_completion_time": 0
    }
  },
  "total_tasks": 0,
  "completed_tasks": 0,
  "delayed_tasks": 0,
  "avg_score": 0.0
}
```

### 3. 작업 (Tasks)

#### 3.1 작업 목록 조회

- **URL:** /api/tasks/
- **Method:** GET
- **Query Parameters:**
  - **status:** TODO/IN\_PROGRESS/REVIEW/DONE/HOLD
  - **priority:** LOW/MEDIUM/HIGH/URGENT
  - **assignee:** 담당자 ID
  - **department:** 부서 ID
  - **search:** 검색어
  - **start\_date:** YYYY-MM-DD
  - **end\_date:** YYYY-MM-DD
- **Response (200):**

```
{
  "count": 0,
  "next": "string(url)",
  "previous": "string(url)",
  "total_pages": 0,
  "current_page": 0,
  "results": [
    {
      "id": 0,
      "title": "string",
      "description": "string",
      "status": "string",
      "priority": "string",
      "assignee": 0,
      "assignee_name": "string",
      "assignee_full_name": "string",
      "reporter": 0,
      "reporter_name": "string",
      "department": 0,
      "department_name": "string",
      "department_parent_id": 0,
      "start_date": "string",
      "due_date": "string",
      "completed_at": "string",
      "created_at": "string",
      "updated_at": "string",
      "estimated_hours": 0.0,
      "actual_hours": 0.0,
      "difficulty": "string",
      "is_delayed": false
    }
  ]
}
```

### 3.2 작업 상세 조회

- **URL:** `/api/tasks/{id}/`
- **Method:** `GET`
- **Response (200):** 작업 상세 정보 (3.1의 응답 형식과 동일)

### 3.3 작업 생성

- **URL:** `/api/tasks/`
- **Method:** `POST`
- **Request Body:** 작업 생성 정보
- **Response (201):** 생성된 작업 정보

### 3.4 작업 수정

- **URL:** `/api/tasks/{id}/`

- **Method:** PATCH
- **Request Body:** 수정할 필드
- **Response (200):** 수정된 작업 정보

### 3.5 캘린더 뷰 작업 목록

- **URL:** /api/tasks/calendar/
- **Method:** GET
- **Query Parameters:**
  - **start\_date:** YYYY-MM-DD
  - **end\_date:** YYYY-MM-DD
  - **assignee:** 담당자 ID (선택)
  - **department:** 부서 ID (선택)
- **Response (200):**

```
[
  {
    "id": 0,
    "title": "string",
    "start_date": "string",
    "due_date": "string",
    "status": "string",
    "priority": "string",
    "is_milestone": false,
    "assignee": 0,
    "is_delayed": false,
    "color": "string",
    "textColor": "string",
    "progress": 0
  }
]
```

### 3.6 작업 통계

- **URL:** /api/tasks/stats/
- **Method:** GET
- **Response (200):**

```
{
  "total": {
    "count": 0,
    "trend": 0.0
  },
  "in_progress": {
    "count": 0,
    "trend": 0.0
  },
  "completed": {
```

```
    "count": 0,  
    "trend": 0.0  
  },  
  "delayed": {  
    "count": 0,  
    "trend": 0.0  
  }  
}
```

### 3.7 팀 성과

- **URL:** `/api/tasks/team-performance/`
- **Method:** `GET`
- **Response (200):**

```
{  
  "members": [  
    {  
      "user_id": 0,  
      "name": "string",  
      "completion_rate": 0.0,  
      "task_count": 0,  
      "average_score": 0.0  
    }  
  ]  
}
```

### 3.8 우선순위별 통계

- **URL:** `/api/tasks/priority-stats/`
- **Method:** `GET`
- **Response (200):**

```
[  
  {  
    "priority": "string",  
    "count": 0,  
    "percentage": 0.0  
  }  
]
```

### 3.9 다가오는 마감일 작업

- **URL:** `/api/tasks/upcoming-deadlines/`
- **Method:** `GET`
- **Response (200):**

```
[
  {
    "id": 0,
    "title": "string",
    "status": "string",
    "priority": "string",
    "due_date": "string",
    "assignee": 0,
    "assignee_name": "string"
  }
]
```

### 3.10 작업 일정 업데이트

- **URL:** `/api/tasks/{id}/update_dates/`
- **Method:** `POST`
- **Request Body:**

```
{
  "start_date": "string",
  "due_date": "string"
}
```

- **Response (200):** 수정된 작업 정보

### 3.11 리소스 할당 상황

- **URL:** `/api/tasks/workload/`
- **Method:** `GET`
- **Query Parameters:**
  - **date:** YYYY-MM-DD
  - **department:** 부서 ID
- **Response (200):**

```
[
  {
    "user_id": 0,
    "user_name": "string",
    "tasks_count": 0
  }
]
```

## 4. 작업 코멘트 (Task Comments)

### 4.1 코멘트 목록 조회



- **URL:** `/api/task-comments/`
- **Method:** `GET`
- **Query Parameters:**
  - `task`: 작업 ID
- **Response (200):**

```
{
  "count": 0,
  "results": [
    {
      "id": 0,
      "task": 0,
      "author": 0,
      "author_name": "string",
      "content": "string",
      "created_at": "string",
      "updated_at": "string"
    }
  ]
}
```

## 4.2 코멘트 작성

- **URL:** `/api/task-comments/`
- **Method:** `POST`
- **Request Body:**

```
{
  "task": 0,
  "content": "string"
}
```

- **Response (201):** 생성된 코멘트 정보

## 5. 작업 첨부파일 (Task Attachments)

### 5.1 첨부파일 목록 조회

- **URL:** `/api/task-attachments/`
- **Method:** `GET`
- **Query Parameters:**
  - `task`: 작업 ID
- **Response (200):**

```
{
  "count": 0,
```

```

    "results": [
      {
        "id": 0,
        "task": 0,
        "file": "string",
        "filename": "string",
        "uploaded_by": 0,
        "uploaded_by_name": "string",
        "created_at": "string"
      }
    ]
  }
}

```

## 5.2 첨부파일 업로드

- **URL:** `/api/task-attachments/`
- **Method:** `POST`
- **Content-Type:** `multipart/form-data`
- **Request Body:**

```

{
  "task": 0,
  "file": "file",
  "filename": "string"
}

```

- **Response (201):** 업로드된 첨부파일 정보

## 6. 작업 평가 (Task Evaluations)

### 6.1 평가 목록 조회

- **URL:** `/api/task-evaluations/`
- **Method:** `GET`
- **Query Parameters:**
  - `task`: 작업 ID
  - `difficulty`: 난이도
- **Response (200):**

```

{
  "count": 0,
  "results": [
    {
      "id": 0,
      "task": {
        "id": 0,
        "title": "string"
      },
    },
  ]
}

```

```

        "evaluator": 0,
        "evaluator_name": "string",
        "difficulty": "string",
        "performance_score": 0,
        "feedback": "string",
        "created_at": "string"
    }
]
}

```

## 6.2 평가 생성

- **URL:** `/api/task-evaluations/`
- **Method:** `POST`
- **Request Body:**

```

{
  "task": 0,
  "difficulty": "string",
  "performance_score": 0,
  "feedback": "string"
}

```

- **Response (201):** 생성된 평가 정보

## 7. 알림 (Notifications)

### 7.1 알림 목록 조회

- **URL:** `/api/notifications/`
- **Method:** `GET`
- **Response (200):**

```

{
  "count": 0,
  "results": [
    {
      "id": 0,
      "recipient": 0,
      "recipient_name": "string",
      "notification_type": "string",
      "task": 0,
      "task_title": "string",
      "message": "string",
      "is_read": false,
      "priority": "string",
      "expires_at": "string",
      "created_at": "string"
    }
  ]
}

```

```
}  
]  
}
```

## 7.2 읽지 않은 알림 개수

- **URL:** `/api/notifications/unread-count/`
- **Method:** `GET`
- **Response (200):**

```
{  
  "count": 0  
}
```

## 7.3 모든 알림 읽음 처리

- **URL:** `/api/notifications/mark-all-read/`
- **Method:** `POST`
- **Response (200):**

```
{  
  "detail": "모든 알림이 읽음 처리되었습니다."  
}
```

# 8. 부서 (Departments)

## 8.1 부서 목록 조회

- **URL:** `/api/departments/`
- **Method:** `GET`
- **Query Parameters:**
  - `parent_isnull`: true/false
- **Response (200):**

```
{  
  "count": 0,  
  "results": [  
    {  
      "id": 0,  
      "name": "string",  
      "code": "string",  
      "parent": 0,  
      "parent_name": "string",  
      "created_at": "string",  
      "updated_at": "string"  
    }  
  ]  
}
```

```
}  
]  
}
```

## 8.2 부서 생성

- **URL:** `/api/departments/`
- **Method:** `POST`
- **Request Body:**

```
{  
  "name": "string",  
  "code": "string",  
  "parent": 0  
}
```

- **Response (201):** 생성된 부서 정보

## 9. 보고서 (Reports)

### 9.1 개인 보고서

- **URL:** `/api/reports/personal-report/`
- **Method:** `GET`
- **Query Parameters:**
  - `start_date`: YYYY-MM-DD
  - `end_date`: YYYY-MM-DD
  - `employee_id`: 직원 ID (선택)
- **Response (200):**

```
{  
  "basic_stats": {  
    "total_tasks": 0,  
    "completed_tasks": 0,  
    "in_progress_tasks": 0,  
    "delayed_tasks": 0  
  },  
  "time_stats": {  
    "average_completion_time": "string",  
    "estimated_vs_actual": 0.0,  
    "daily_work_hours": []  
  },  
  "quality_stats": {  
    "average_score": 0.0,  
    "review_rejection_rate": 0.0,  
    "rework_rate": 0.0  
  },  
  "distribution_stats": {
```

```

    "priority_distribution": [],
    "difficulty_distribution": [],
    "status_distribution": []
  },
  "comparison_stats": {
    "team_comparison": {
      "team_avg_completion_time": "string",
      "team_avg_score": 0.0,
      "my_completion_time": "string",
      "my_score": 0.0,
      "relative_efficiency": 0.0
    },
    "department_comparison": {
      "dept_avg_completion_time": "string",
      "dept_avg_score": 0.0,
      "my_completion_time": "string",
      "my_score": 0.0,
      "relative_efficiency": 0.0
    }
  }
}

```

## 10. 활동 로그 (Activities)

### 10.1 활동 로그 조회

- **URL:** `/api/activities/`
- **Method:** `GET`
- **Response (200):**

```

{
  "count": 0,
  "results": [
    {
      "id": 0,
      "type": "string",
      "description": "string",
      "created_at": "string",
      "task_id": 0,
      "task_title": "string"
    }
  ]
}

```

### 10.2 최근 활동 조회

- **URL:** `/api/activities/recent/`
- **Method:** `GET`
- **Response (200):**

```
[
  {
    "id": 0,
    "type": "string",
    "description": "string",
    "created_at": "string",
    "task_id": 0,
    "task_title": "string"
  }
]
```

## 11. 실험 (Experiments)

### 11.1 LLM 분석

- **URL:** `/api/experiments/llm/analyze/`
- **Method:** `POST`
- **Request Body:**

```
{
  "question": "string"
}
```

- **Response (200):**

```
{
  "question": "string",
  "sql_query": "string",
  "result": "any",
  "formatted_result": "string"
}
```

## 12. 공통 사항

### 12.1 인증

- 모든 API 요청에 Authorization 헤더 필요

```
Authorization: Bearer {access_token}
```

### 12.2 페이지네이션

- 기본 페이지 크기: 10
- 최대 페이지 크기: 100

- Query Parameters:
  - **page**: 페이지 번호
  - **page\_size**: 페이지당 항목 수

## 12.3 에러 응답

### 인증 오류 (401)

```
{
  "detail": "인증 자격 증명이 제공되지 않았습니다."
}
```

### 권한 오류 (403)

```
{
  "detail": "이 작업을 수행할 권한이 없습니다."
}
```

### 유효성 검사 오류 (400)

```
{
  "field_name": ["에러 메시지"]
}
```

## 권한 체계

---

### 사용자 권한 레벨

#### Role (역할)

- ADMIN: 최고 관리자
- MANAGER(Role): 관리자(팀장)
- EMPLOYEE: 일반 직원

#### Rank (직급)

- DIRECTOR: 이사
- GENERAL\_MANAGER: 본부장
- DEPUTY\_GENERAL\_MANAGER: 차장 --> 현재 안쓰이고 있음.
- MANAGER(Rank): 과장 (팀장 -> 현재는 팀장으로 통일)
- ASSISTANT\_MANAGER: 대리
- SENIOR: 주임



- STAFF: 사원

## API 접근 권한

### Tasks API

#### 작업 조회 (/api/tasks/)

- 목록 조회: 모든 사용자 접근 가능
- 상세 조회:
  - ADMIN(Role): 모든 작업 접근 가능
  - DIRECTOR/GENERAL\_MANAGER(Rank): 본부 (본부 + 산하 팀) 내 모든 작업 접근 가능
  - MANAGER(Role): 팀 내 작업만 접근 가능
  - EMPLOYEE(Role): 자신의 작업만 접근 가능

### Users API

#### 사용자 관리 (/api/users/)

- 목록 조회: 모든 사용자 접근 가능
- 상세 조회/수정:
  - ADMIN(Role): 모든 직원 정보 접근/수정 가능
  - DIRECTOR/GENERAL\_MANAGER(Rank): 모든 직원 정보 접근/수정 가능
  - MANAGER/EMPLOYEE(Role): 자신의 정보만 접근 가능

### Reports API

#### 보고서 조회 (/api/reports/)

- ADMIN(Role): 모든 보고서 접근 가능
- DIRECTOR/GENERAL\_MANAGER(Rank): 모든 보고서 접근 가능
- MANAGER(Role): 팀 보고서만 접근 가능
- EMPLOYEE(Role): 자신의 보고서만 접근 가능

### Evaluations API

#### 작업 평가 (/api/task-evaluations/)

- ADMIN(Role): 모든 작업 평가 가능
- DIRECTOR/GENERAL\_MANAGER(Rank): 본부 (본부 + 산하 팀) 내의 직원들 모두에 작업 평가 가능
- MANAGER(Role): 팀 내 작업만 평가 가능
- EMPLOYEE(Role): 평가 권한 없음

## 권한별 주요 기능

### 일반 직원 (EMPLOYEE Role)

- 모든 작업 목록 조회 가능
- 자신의 작업만 상세 조회/수정 가능

- 모든 직원 목록 조회 가능
- 자신의 정보만 상세 조회/수정 가능
- 자신의 보고서만 조회 가능
- 자신의 통계 정보만 조회 가능
- 자신의 알림만 조회 가능
- 작업 코멘트 작성/조회 가능
- 작업 시간 로그 기록 가능
- 평가 권한 없음

### 팀장 (MANAGER Role)

- 모든 작업 목록 조회 가능
- 팀 내 작업만 상세 조회/관리 가능
- 모든 직원 목록 조회 가능
- 자신의 정보만 상세 조회/수정 가능
- 팀 보고서 조회 가능
- 팀원들의 통계 정보 조회 가능
- 팀 내 작업 검토 요청 처리
- 팀원 작업 평가 가능
- 팀 내 작업 배정 가능

### 본부장/이사 (GENERAL\_MANAGER/DIRECTOR Rank)

- 모든 작업 목록 조회 가능
- 본부 (본부 + 산하 팀) 내 작업 상세 조회/관리 가능
- 모든 직원 목록 조회 가능
- 모든 직원 정보 상세 조회/수정 가능
- 모든 보고서 접근 가능
- 본부 전체 통계 조회 가능
- 직원 등록/수정/삭제 권한
- 부서 생성/수정/삭제 권한 (이사만)
- 본부 내 작업 배정 권한
- 본부 내 직원들의 작업 평가 가능

### 최고 관리자 (ADMIN Role)

- 모든 작업 목록/상세 조회 및 관리 가능
- 모든 직원 목록/상세 조회 및 관리 가능
- 모든 보고서 접근 가능
- 모든 통계 조회 가능
- 시스템 설정 관리
- 모든 데이터 조회/수정/삭제 가능
- 모든 작업 평가 가능