

Software Engineering Project Team 25

Milestone 5

Sync'em

An AI-powered assistant app that helps HR and project managers efficiently allocate projects based on employee skills and availability, while providing a chatbot for employees for quick HR policy queries.

Team Members

1. AKBAR ALI | 23f1002997
2. DEON LEVON DMELLO | 21f3002473
3. D NARENDRAN | 22f3002698
4. SAMPRITI RAHA | 22f1000064
5. SAYAN BHOWMICK | 22f3001657
6. SOHAM CHAKRABORTY | 23f1001783
7. TELVIN VARGHESE | 22f2001640
8. TUMMALA NAVNEEN | 22f3002759

Table of Contents

1. Admin APIs
2. Employee APIs
3. Project Manager APIs
4. Human Resource APIs

1. Admin APIs

1.1

Route/Path: /api/admin/employees

Request Method: GET

Inputs:

JSON: not applicable

Header: {"Authorization": f"Bearer {admin_token}"}

admin/employees

Expected Output:

HTTP Status Code: 200

JSON: List of All the employees, including the current user, in a JSON format.

Actual Output:

HTTP Status Code: 200

JSON:

```
{  "id": 2,  "name": "pm",
  "email": "pm@gmail.com",  "role": "product_manager"},

{  "id": 3,  "name": "hr",
  "email": "hr@gmail.com",  "role": "human_resource"},

{  "id": 5,  "name": "Admin",
  "email": "admin429@gmail.com",  "role": "root"},

{  "id": 4,  "name": "New Automated Test Name",
  "email": "emp@gmail.com",  "role": "employee"},

{  "id": 1,  "name": "Root",
  "email": "root@gmail.com",  "role": "root"},

{  "id": 7,  "name": "Admin",
  "email": "admin678@gmail.com",  "role": "root"}
```

Result: Success

1.2

Route/Path: /api/admin/account

Request Method: GET

Inputs:

JSON: not applicable

Header: {"Authorization": f"Bearer {admin_token}"}

admin/account

Expected Output:

HTTP Status Code: 200

JSON: List having id, name, role etc for the current user (admin)

Actual Output:

HTTP Status Code: 200

JSON:

```
{ "id": 1, "name": "Root",
  "email": "root@gmail.com", "role": "root" }
```

Result: Success

2. Employees APIs

2.1

Route/Path: /api/employee/dashboard

Request Method:

Input: not applicable

Header Token: {"Authorization": f"Bearer {employee_token}"}

employee/dashboard

Expected Output:

HTTP Status Code: 200

JSON: List of all message, stats, requests, tasks, announcements of the Employee

Actual Output:

HTTP Status Code: 200

JSON:

```
{ "message": "Dashboard data retrieved successfully",
  "stats": { "pending_tasks": 1, "completed_tasks": 1,
  "requests": 34, "courses_completed": 0 },
  "tasks": [ { "id": 4, "task": "Updated task",
    "status": "completed", "deadline": null,
    "date_created": "2025-11-30T17:33:45.307489" },
  { "id": 2, "task": "Buy office supplies",
    "status": "pending", "deadline": null,
    "date_created": "2025-11-30T17:11:48.691191" } ],
  "announcements": [ { "id": 2,
    "announcement": "Office will remain closed on Friday",
    "created_at": "2025-11-30T17:12:24.051698" } ],
  "user": { "id": 4, "name": "New Automated Test Name",
    "email": "emp@gmail.com", "role": "employee" } }
```

Result: Success

2.2

Route/Path: /api/employee/requests/leave

Request Method: POST

Input:

```
{
  "leave_type": "Sick Leave",
  "from_date": "2025-01-20T00:00:00",
  "to_date": "2025-01-22T00:00:00",
  "reason": "Fever & cold"
}
```

Header Token: {"Authorization": f"Bearer {employee_token}"}

[employee/request/leave](#)

Expected Output:

HTTP Status Code: 200

JSON: List of the message, request_id and leave_id

Actual Output:

HTTP Status Code: 200

JSON:

```
{  "message": "Leave request submitted",
  "request_id": 37,  "leave_id": 23  }
```

Result: Success

3. Project Manager APIs

3.1

Route/Path: /api/pm/dashboard

Request Method: GET

Input: not applicable

Header Token: {"Authorization": f"Bearer {pm_token}"}

[pm/dashboard](#)

Expected Output:

HTTP Status Code: 200

JSON: List of clients, projects and stats

Actual Output:

HTTP Status Code: 200

JSON:

```
{  "message": "Dashboard data retrieved successfully",
  "data":
    {  "user":
      {  "id": 2,  "name": "pm",
        "email": "pm@gmail.com",  "role": "product_manager" },
      "ClientList": [
        {  "id": 1,
          "clientname": "Test Client Pvt Ltd",
          "description": "dGVzdCBkZXJhawWw=" },
        {  "id": 2,  "clientname": "Test Client Pvt Ltd",
```

```

"description": "dGVzdCBkZXRhaWw=" },
],

"projects": [
{ "id": 2, "project_id": "P3892",
"project_name": "Updated Project Pvt Ltd",
"description": "Updated Project Pvt Ltd",
"status": "PENDING",
"client_id": 2, "manager_id": 1 },

{ "id": 4, "project_id": "P6177",
"project_name": "Test Project Pvt Ltd",
"description": "Test Project Pvt Ltd",
"status": "PENDING", "client_id": 10, "manager_id": 1 }],
}

"stats":
{ "total_projects": 4, "active_projects": 0,
"completed_projects": 0, "pending_projects": 4 } } }

```

Result: Success

3.2

Route/Path: /api/pm/employee/performance/1

Request Method: GET

Input: not applicable

Header Token: {"Authorization": f"Bearer {pm_token}"}

[pm/employee/performance](#)

Expected Output:

HTTP Status Code: 200

JSON: List of message, employee and their current stats and performance trends

Actual Output:

HTTP Status Code: 200

JSON:

```

{ "message": "Employee performance retrieved successfully",
"data":
{ "employee": 
```

```
{ "id": 1, "name": "Updated Name",  
"email": "root@gmail.com", "role": "root" },  
  
"current_stats": { "completed": 0, "in_progress": 0,  
"pending": 0, "total": 0, "completed_percentage": 0,  
"in_progress_percentage": 0, "pending_percentage": 0 },  
  
"performance_trends": [  
  { "month": "Jan", "score": 65 },  
  { "month": "Feb", "score": 72 },  
  { "month": "Mar", "score": 68 },  
  { "month": "Apr", "score": 78 },  
  { "month": "May", "score": 85 },  
  { "month": "Jun", "score": 88 }  
]
```

Result: Success

4. Human Resource APIs

4.1

Route/Path: /api/hr/course

Request Method: GET

Input: not applicable

Header Token: {"Authorization": f"Bearer {hr_token}"}

[hr/course](#)

Expected Output:

HTTP Status Code: 200

JSON: List of all courses available

Actual Output:

HTTP Status Code: 200

JSON:

```
[  
  { "id": 9,  
    "course_name": "Duplicate Assignment Test Course",  
    "instructor": "John Doe",  
    "start_date": "2023-09-01",  
    "end_date": "2023-10-31",  
    "description": "A test course for duplicate assignments.",  
    "status": "Active"  
}
```

```
"course_link": "https://example.com/dup-course",
"topics": "Duplicate, Testing"  },

{  "id": 8,
"course_name": "Test Course for Assignment",
"course_link": "https://example.com/test-course",
"topics": "Testing, Assignment"  },

{  "id": 6,
"course_name": "Duplicate Assignment Test Course",
"course_link": "https://example.com/dup-course",
"topics": "Duplicate, Testing"  }
]
```

Result: Success

4.2

Route/Path: /api/hr/policies

Request Method: POST

Input:

```
{
"id":"1",
"title":"first quality check",
"content":"anything"
}
```

Header Token: {"Authorization": f"Bearer {pm_token}"}

[hr/policies](#)

Expected Output:

HTTP Status Code: 200

JSON: List of message and policy information

Actual Output:

HTTP Status Code: 200

JSON:

```
{  
    "message": "Policy created",  
    "policy": {  
        "created_at": "2025-11-30T18:35:00.615982",  
        "id": 4,  
        "title": "first quality check",  
        "content": "anything"  
    }  
}
```

Result: Success

5. Other Testcase

5.1 Where the expected output does not actual output

Route/Path: /api/employee/requests/reimbursement

Request Method: POST

Input:

```
{  
    "expense_type": "Travel",  
    "amount": "500$",  
    "date_expense": "2025-01-22T00:00:00",  
    "remark": "Taxi",  
}
```

Header Token: {"Authorization": f"Bearer {emp_token}"}

[request/reimbursement](#)

Expected Output:

HTTP Status Code: 200

JSON: amount reimbursed

Actual Output:

HTTP Status Code: 422

JSON:

```
{ "detail": [
{ "type": "json_invalid",
"loc": [ "body", 106 ],
"msg": "JSON decode error",
"input": {},
"context": { "error": "Illegal trailing
comma before end of object" } } ] }
```

Result: Failed