Case Management: Leave Approval Page – Data Loading Issue

User → Frontend Page → Application Server → Database (Stored Procedure) → Database Tables

1. Overview

Users reported that the **Leave Approval Page** is taking too long to load data in the grid. While the data is eventually fetched correctly, the **timely interactive experience** is affected, causing delays in processing leave approvals.

2. User Interaction Layer (Frontend)

Observation: When users visit the Leave Approval Page, the grid/table shows a loading indicator for a long time.

Displayed Columns:

1. Employee Name
2. Leave Name
3. Applied On Date
4. From Date
5. To Date
6. Leave Taken Days
7. Joining Date
8. Approve/Reject
9. Report

Impact: Slow frontend rendering due to delayed data retrieval from backend.

3. Application Server Layer (API/Backend)

Inspection: The API that fetches data for the Leave Approval Page was analyzed.

Observation: The API correctly calls the backend stored procedure (SP) to fetch leave data.

Impact: The API is dependent on the database SP execution time; the bottleneck is not the API itself but the SP.

4. Database Layer (Stored Procedure)

Observation:

The main logic for fetching leave data resides in a **stored procedure**.

The SP performs filtering across multiple columns (Employee, Leave Type, Dates, Status, etc.).

The SP currently performs full table scans due to lack of optimized filtering and indexing.

Root Cause:

The SP is **not optimized for performance**.

Filtering on all columns and joining multiple tables without proper indexing results in **slow query execution**.

Recommended Action:

Optimize SP logic (rewrite inefficient joins, remove unnecessary subqueries).

Limit columns to only those required by the frontend.

Add appropriate indexes on frequently filtered columns (DepartmentID, Status, AppliedDate, etc.).

Consider **pagination** or **set-based operations** to reduce data load.

5. Database Tables

Observation:

Leave data is stored in multiple tables (Leaves, Employees, Departments).

Current queries fetch large datasets unnecessarily.

Impact: Large data volume contributes to slow SP execution and delayed frontend loading.

Conclusion:
The primary bottleneck causing slow data loading on the Leave Approval Page is the **stored procedure at the database layer**. Optimizing the SP will significantly improve performance and provide a timely interactive experience for users.