The objective: of this test assignment is to develop a grid system with filtering functionality in the backend. The grid system is designed to display student details(id, name , total marks etc) and allow filtering based on various columns. The assignment involves developing the backend APIs responsible for loading student details with pagination, and implementing server-side filtering.

Requirements: Load Student Details API: Implement an API that retrieves student details from a file(csv/json / any other format) and returns the data in a paginated manner. The API should accept parameters such as page number and page size to allow pagination. Server-side Filtering API: Implement server-side filtering functionality in the backend. The filtering mechanism should allow the UI to send filter criteria to the backend API, which will then return the filtered results to the UI.

I will write code on SQL

1. Load Student Details API (Pagination):

CREATE TABLE students (

id INT PRIMARY KEY,

name VARCHAR(50),

total\_marks INT

);

-- Assuming the data is already loaded into the "students" table

-- API to retrieve paginated student details

CREATE PROCEDURE GetPaginatedStudents(

@PageNumber INT,

@PageSize INT

)

AS

BEGIN

SELECT id, name, total\_marks

FROM (

SELECT ROW\_NUMBER() OVER (ORDER BY id) AS RowNum, id, name, total\_marks

FROM students

) AS Sub

WHERE RowNum >= (@PageNumber - 1) \* @PageSize + 1

AND RowNum <= @PageNumber \* @PageSize;

END;

1. Server-side Filtering API:

-- API to retrieve filtered student details

CREATE PROCEDURE GetFilteredStudents(

@FilterColumn VARCHAR(50),

@FilterValue VARCHAR(50)

)

AS

BEGIN

DECLARE @FilterQuery VARCHAR(MAX);

SET @FilterQuery = 'SELECT id, name, total\_marks FROM students WHERE ' + @FilterColumn + ' = @FilterValue;';

EXECUTE sp\_executesql @FilterQuery, N'@FilterValue VARCHAR(50)', @FilterValue;

END;