**Backend API Technical Requirement Document**

**BOOKS MANAGEMENT SYSTEM – REGISTER API**

**1. Overview**

This document outlines the requirements for creating backend API endpoints to handle user registration for the Books Management System.

**2. Objective**

* Develop API endpoints for user registration.
* Implement validation for user registration data.
* Integrate the backend with the database to store user information.

**3. Success Criteria**

* API returns a success response upon successful user registration.
* API returns an error response upon invalid user registration data.
* User registration data is securely stored in the database.

**4. Endpoints**

* **POST /api/auth/register**
  + **Purpose:** Register a new user.
  + **Request Body:**

Json { "username": "string", "password": "string", "email": "string", "gender": "string", "age": "number", "contactNumber": "string" }

* + **Response:**
    - Success (HTTP 201 Created):

Json { "success": true, "message": "User registered successfully" }

* + - Error (HTTP 400 Bad Request):

Json { "success": false, "message": "Validation error: Invalid data provided" }

**5. Validation**

* Validate that all required fields are provided in the request body.
* Validate the format of email address.
* Validate the format of contact number.
* Ensure username is unique in the database.

**6. Error Handling**

* Invalid data provided: Return HTTP 400 Bad Request status code with an appropriate error message.
* Server errors: Return HTTP 500 Internal Server Error status code with an error message.

**7. Data Storage**

* Store user registration data securely in the database.
* Hash passwords before storing them to ensure security.
* Implement proper database indexes and constraints to ensure data integrity.