Just Built a Bulk Student Import System with Node.js & Vue.js! and MongoDB
I'm excited to share this new feature I developed for our school management system:

NPM PACKAGES: npm install xlsx csv-parser multer

Backend:
Model-

```
// models/studentModel.js
"use strict";

const mongoose = require('mongoose');

const studentSchema = new mongoose.Schema({
    firstName: { type: String, required: true },
    lastName: { type: String, required: true },
    admissionNumber: {
        type: String,
        required: true,
        unique: true
    },
    phoneNumber: { type: String, required: true },
    createdAt: { type: Date, default: Date.now }
});

const Student = mongoose.model('StudentData', studentSchema);

module.exports = Student;
```

Routes:

```
// routes/studentRoutes.js
"use strict";

const express = require('express');
const multer = require('multer');
const { importStudents } = require('../services/studentservice');
const router = express.Router();
const upload = multer({ storage: multer.memoryStorage() });
const prefix = '/api/studentsImport'

router.post(`${prefix}/import`, upload.single('file'), async (req, res) => {
    console.log('API HIT');

    try {
        if (!req.file) {
            return res.status(400).json({
                success: false,
                message: 'No file uploaded'
            });
        }

        const result = await importStudents(req.file.buffer);
        res.json(result);
    } catch (error) {
        res.status(500).json({
            success: false,
            message: error.message
        });
    }
});

module.exports = router;
```

SERVICE:

```
// services/studentImportService.js
"use strict";

const xlsx = require('xlsx');
const Student = require('../models/student');

async function importStudents(fileBuffer) {
    try {
        // Parse Excel file
        const workbook = xlsx.read(fileBuffer);
        const worksheet = workbook.Sheets[workbook.SheetNames[0]];
        const jsonData = xlsx.utils.sheet_to_json(worksheet);

        // Transform data
        const students = jsonData.map(row => ({
            firstName: row.FirstName,
            lastName: row.LastName,
            admissionNumber: row.AdmissionNumber,
            phoneNumber: row.PhoneNumber.toString() // Ensure string format
        }));
        console.log('students', students);

        // Validate required fields
        for (const student of students) {
            if (!student.firstName || !student.lastName || !student.admissionNumber ||
!student.phoneNumber) {
                throw new Error('Missing required fields in one or more rows');
            }
        }

        // Save to database
        const result = await Student.insertMany(students, { ordered: false });
        return {
            success: true,
            importedCount: result.length,
            totalCount: jsonData.length
        };
    } catch (error) {
        if (error.writeErrors) {
            const duplicates = error.writeErrors.map(err => err.err.op.admissionNumber);
            return {
                success: false,
                message: `Duplicate admission numbers: ${duplicates.join(', ')}`,
                importedCount: error.result.nInserted,
                totalCount: jsonData.length
            };
        }
        throw error;
    }
}

module.exports = { importStudents };
```

FRONTEND:(vuejs)

```
<template>
    <div class="import-container">
        <h3>Bulk Student Import</h3>

        <!-- Download Template -->
        <div class="mb-4">
            <VBtn @click="downloadTemplate" class="btn btn-secondary">
                Download Template
            </VBtn>
        </div>

        <!-- File Upload -->
        <div class="file-upload">
            <input type="file" ref="fileInput" @change="handleFileChange" accept=".xlsx, .xls, .csv"
                class="form-control" />
            <VBtn @click="submitFile" :disabled="!file" class="btn btn-primary mt-2">
                Import Students
            </VBtn>
        </div>

        <!-- Results -->
        <div v-if="importResult" class="mt-4">
            <div v-if="importResult.success" class="alert alert-success">
```

```
                Successfully imported {{ importResult.importedCount }} of {{ importResult.totalCount }} students
            </div>
            <div v-else class="alert alert-danger">
                {{ importResult.message }}<br>
                Imported {{ importResult.importedCount }} of {{ importResult.totalCount }} students
            </div>
        </div>
    </div>
</template>

<script setup>
import axios from 'axios';
import { ref } from 'vue';
import * as XLSX from 'xlsx';

const fileInput = ref(null);
const file = ref(null);
const importResult = ref(null);

const handleFileChange = (e) => {
    file.value = e.target.files[0];
};

const downloadTemplate = () => {
    const templateData = [
        {
            'FirstName': 'John',
            'LastName': 'Doe',
            'AdmissionNumber': 'ADM001',
            'PhoneNumber': '1234567890'
        },
        {
            'FirstName': 'Jane',
            'LastName': 'Smith',
            'AdmissionNumber': 'ADM002',
            'PhoneNumber': '9876543210'
        },
        {
            'FirstName': 'Pk',
            'LastName': 'Doe',
            'AdmissionNumber': 'ADM001',
            'PhoneNumber': '1234567891'
        },
        {
            'FirstName': 'gopi',
            'LastName': 'Smith',
            'AdmissionNumber': 'ADM002',
            'PhoneNumber': '9876543212'
        }
    ];

    const ws = XLSX.utils.json_to_sheet(templateData);
    const wb = XLSX.utils.book_new();
    XLSX.utils.book_append_sheet(wb, ws, "Students");
    XLSX.writeFile(wb, "student_import_template.xlsx");
};

const submitFile = async () => {
    if (!file.value) return;

    const formData = new FormData();
    formData.append('file', file.value);

    try {
        const response = await axios.post('http://127.0.0.1:5000/api/studentsImport/import',
formData, {
            headers: {
                'Content-Type': 'multipart/form-data'
            }
        });
        importResult.value = response.data;
        fileInput.value.value = ''; // Reset file input
        file.value = null;
    } catch (error) {
        console.log('error', error);
        console.log('error', error.response);

        importResult.value = {
            success: false,
            message: error.response?.data?.message || 'Import failed',
            importedCount: error.response?.data?.importedCount || 0,
            totalCount: error.response?.data?.totalCount || 0
        };
    }
};
</script>


<style scoped>
.import-container {
    max-width: 600px;
    margin: 0 auto;
    padding: 20px;
}
```

```
.alert {
    padding: 15px;
    border-radius: 4px;
}

.alert-success {
    background-color: #d4edda;
    color: #155724;
}

.alert-danger {
    background-color: #f8d7da;
    color: #721c24;
}
</style>
```