

# React JS Assignment

## OVERVIEW

We have created a basic react application including crud operations with ASP.NET Core 3.1 Ado.Net web API.

## NPM LIBRARIES

1. React
2. Reactstrap
3. Node-SASS (Style)
4. Axios (Http requests)

## NUGET PACKAGES

1. NLog
2. SqlClient

## Student Interface

The screenshot shows a web application interface for managing students. At the top, there is a green navigation bar with a home icon and the text 'Students'. Below this, there is a section titled 'Student Details' with a dropdown arrow. This section contains four input fields: 'First Name', 'Last Name', 'Contact Person', and 'Contact No.'. Below these fields are three buttons: 'Save' (green), 'Delete' (red), and 'Reset' (yellow). Below the 'Student Details' section is another section titled 'Existing Students' with a dropdown arrow. This section contains a table with the following data:

First Name	Last Name	Contact Person	Contact No.
Mario	Speedwagon	Petey Cruiser	771234567
Anna	Sthesis	Paul Molve	771230456
Walter	Melon	Barb Ackue	784561234
Dinushka	Rukshan	Lakshika Madushan	770448899

## TASKS

Please develop CRUD (Create, Insert, Update, Delete) operation.

**1. Develop Students Interface with the following details. (Follow given sample structure for the student interface)**

Students has following details (All \* fields are required)

- Student ID \* (No need to display in the interface)
- First Name \*
- Last Name \*
- Contact Person \*
- Contact No \*
- Email Address \*
- Date of birth \*
- Age \* (Should be auto calculated after entering the DOB)
- Classroom \* (Should be a drop-down box and should load from the classroomstable)

**2. Develop the Classrooms interface with the following details. (Follow the same sample structure in the student interface)**

Classrooms has following details (All \* fields are required)

1. Classroom ID \* (No need to display in the interface)
2. Classroom Name \*

**3. Develop the Teachers interface with the following details. (Follow the same sample structure in the student interface)**

Teachers has following details (All \* fields are required)

1. Teacher ID \* (No need to display in the interface)
2. First Name \*
3. Last Name
4. Contact No \*
5. Email Address \*

**4. Develop the Subjects interface with the following details. (Follow the same sample structure in the student interface)**

Subjects has following details (All \* fields are required)

1. Subject ID \* (No need to display in the interface)
2. Subject Name \*

5. Develop the Allocate Subjects interface with the following details. (Follow the below mockup when developing the interface & should be able to allocate subjects to the teachers)

Allocate subjects has following details (All \* fields are required)

1. AllocateSubject ID \* (No need to display in the interface)
2. Teacher \* (Should be a drop-down box and should load from the teachers table)
3. Subject \* (Should be a drop-down box and should load from the subjects table & able to add to a table(grid))

Allocate Subjects

Teachers details

Teacher

Allocated subjects

Subject

Subjects	Action
English	<input type="button" value="Deallocate"/>
Science	<input type="button" value="Deallocate"/>

Note - Teachers can have multiple subjects.

6. Develop the Allocate Classrooms interface with the following details. (Follow the below mockup when developing the interface & should be able to allocate teachers to the classrooms)

Allocate classrooms has following details (All \* fields are required)

1. AllocateClassroom ID \* (No need to display in the interface)
2. Teacher \* (Should be a drop-down box and should load from the teachers table)
3. Classroom \* (Should be a drop-down box and should load from the classrooms table & able to add to a table(grid))

Allocate Classrooms

Teachers details

Teacher

Select a teacher

Save

Allocated Classrooms

Classroom

Select a classroom

Allocate

Classrooms	Action
Grade 5A	<div>Deallocate</div>
Grade 5B	<div>Deallocate</div>
Grade 6A	<div>Deallocate</div>

Note - Teachers can have multiple classrooms.

7. Develop the Student Detail Report interface with the following details. (Follow the below mockup when developing the interface)

Student Detail Report

Student Details

Student

Asela Rathnayake

Classroom

Grade 5A

Contact Person

Suranga Rathnayake

Email address

aselarathnayake@gmail.com

Contact No.

0771234567

Date of birth

1998-09-02

Teacher & Subject Details

Subject	Teacher
English	Sudarshani bandana
Science	Amila Kuruppu

Copyright © Jay kay Marketing Services (Pvt) Limited

Created By JMSL

Student Detail Report has following details (All \* fields are required)

1. Student \* (Should be a drop-down box and should load from the students table)
2. Contact person
3. Contact No.
4. Email address
5. Date of birth
6. Classroom (Display by linking the classroom table)
7. Subject & teacher list (Should be a table (grid) and should load by linking the relevant tables)

## CONSIDER THE FOLLOWINGS

- User interface design
- Code / Components reuse
- Validations & Exception handling
- React routing
- React state management
- Database design (primary keys, foreign keys, unique constraints, table joins & etc..)