

Perfect! Let's build a Django + React.js full CRUD app for managing students, step-by-step.

We'll cover:

- Backend (Django + DRF):
 - Setup
 - Create student model
 - API endpoints (CRUD)

 - Frontend (React.js + Axios):
 - Show student list
 - Add student
 - Edit student
 - Delete student
-

□ Backend (Django + Django REST Framework)

□ 1. Install & Create Django Project

```
django-admin startproject backend
cd backend
pip install django djangorestframework corsheaders
python manage.py startapp students
```

□ 2. Configure Django `settings.py`

```
INSTALLED_APPS = [
    ...
    'rest_framework',
    'corsheaders',
    'students',
]

MIDDLEWARE = [
    'corsheaders.middleware.CorsMiddleware',
    ...
]

CORS_ALLOW_ALL_ORIGINS = True
```

□ 3. Create Student Model

In `students/models.py`:

```
from django.db import models

class Student(models.Model):
    name = models.CharField(max_length=100)
    age = models.IntegerField()
    grade = models.CharField(max_length=10)

    def __str__(self):
        return self.name
```

Migrate the model:

```
python manage.py makemigrations
python manage.py migrate
```

□ 4. Create Serializer

In `students/serializers.py`:

```
from rest_framework import serializers
from .models import Student

class StudentSerializer(serializers.ModelSerializer):
    class Meta:
        model = Student
        fields = '__all__'
```

□ 5. Create CRUD Views

In `students/views.py`:

```
from rest_framework import viewsets
from .models import Student
from .serializers import StudentSerializer

class StudentViewSet(viewsets.ModelViewSet):
    queryset = Student.objects.all()
    serializer_class = StudentSerializer
```

□ 6. Add URLs

Create `students/urls.py`:

```
from django.urls import path, include
from rest_framework.routers import DefaultRouter
from .views import StudentViewSet

router = DefaultRouter()
router.register(r'students', StudentViewSet)

urlpatterns = [
    path('', include(router.urls)),
]
```

Update backend/urls.py:

```
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('api/', include('students.urls')),
]
```

□ Frontend (React.js)

□ 1. Create React App

```
npx create-react-app frontend
cd frontend
npm install axios
```

□ 2. Axios API Service

Create src/api.js:

```
import axios from 'axios';

export default axios.create({
  baseURL: 'http://127.0.0.1:8000/api/'
});
```

□ 3. Create Student CRUD Component

Create src/components/StudentApp.js:

```
import React, { useEffect, useState } from 'react';
import api from '../api';

const StudentApp = () => {
  const [students, setStudents] = useState([]);
```

```

const [form, setForm] = useState({ name: '', age: '', grade: '' });
const [editId, setEditId] = useState(null);

useEffect(() => {
  fetchStudents();
}, []);

const fetchStudents = async () => {
  const res = await api.get('students/');
  setStudents(res.data);
};

const handleChange = (e) => {
  setForm({ ...form, [e.target.name]: e.target.value });
};

const handleSubmit = async () => {
  if (editId) {
    await api.put(`students/${editId}/`, form);
    setEditId(null);
  } else {
    await api.post('students/', form);
  }
  setForm({ name: '', age: '', grade: '' });
  fetchStudents();
};

const handleEdit = (student) => {
  setForm({ name: student.name, age: student.age, grade: student.grade });
  setEditId(student.id);
};

const handleDelete = async (id) => {
  await api.delete(`students/${id}/`);
  fetchStudents();
};

return (
  <div style={{ padding: '20px' }}>
    <h2>Student Manager</h2>
    <input name="name" value={form.name} onChange={handleChange}
placeholder="Name" />
    <input name="age" type="number" value={form.age}
onChange={handleChange} placeholder="Age" />
    <input name="grade" value={form.grade} onChange={handleChange}
placeholder="Grade" />
    <button onClick={handleSubmit}>{editId ? 'Update' : 'Add'}</button>

    <ul>
      {students.map((s) => (
        <li key={s.id}>
          {s.name} (Age: {s.age}, Grade: {s.grade})
          <button onClick={() => handleEdit(s)}>Edit</button>
          <button onClick={() => handleDelete(s.id)}>Delete</button>
        </li>
      ))}
    </ul>
  </div>
)

```

```
        </div>
    );
};

export default StudentApp;
```

□ 4. Update Main App Component

In `src/App.js`:

```
import React from 'react';
import StudentApp from '../components/StudentApp';

function App() {
  return (
    <div className="App">
      <StudentApp />
    </div>
  );
}

export default App;
```

□ 5. Run Servers

Run Django backend:

```
python manage.py runserver
```

Run React frontend:

```
npm start
```

□ Done!

You now have a full working **Student CRUD App** using:

- Django backend with REST API
- React frontend with Axios