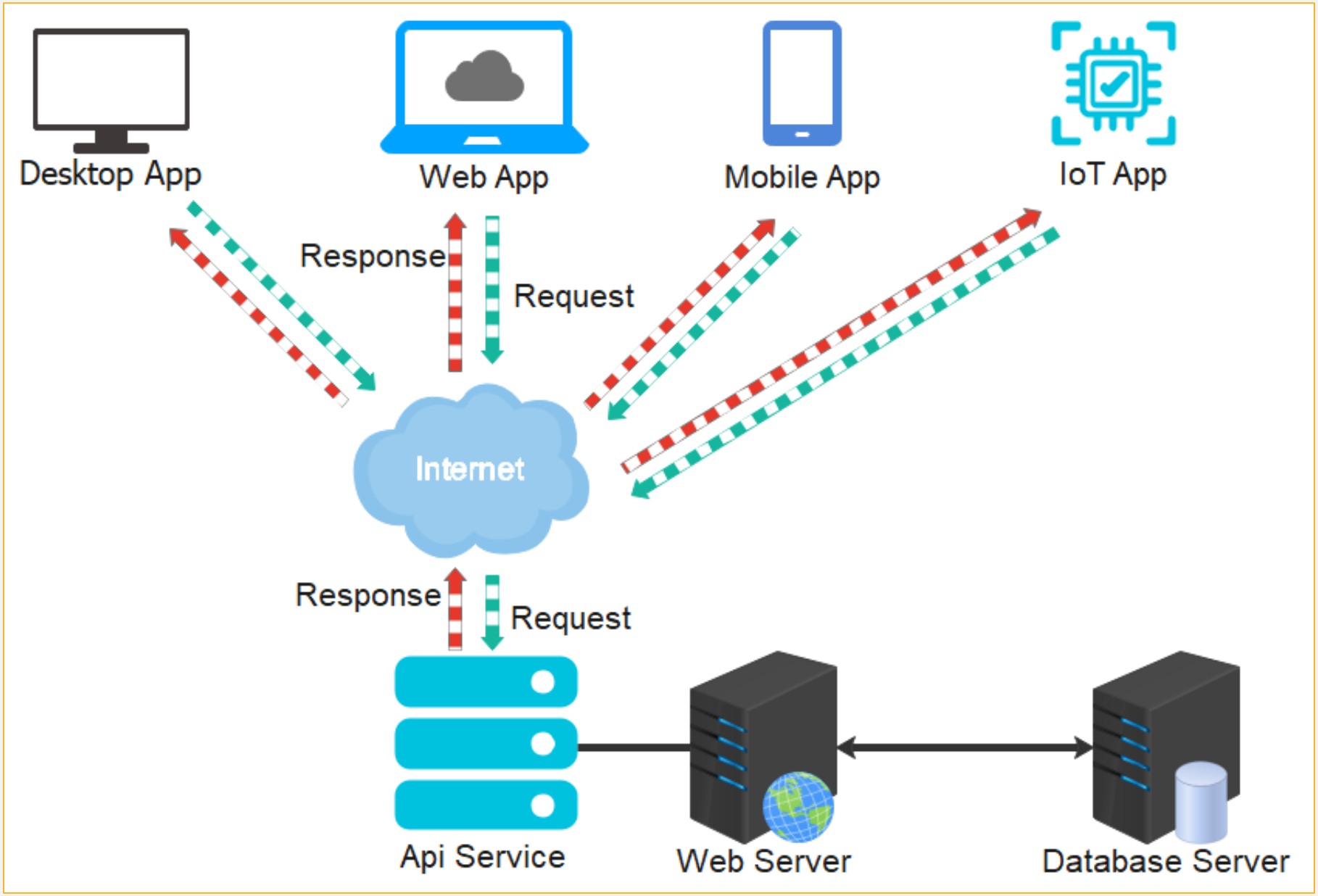
React

Working with API

How the API works



Installing & Configuring Axios

- In terminal, type: npm i axios
- Create a new folder named “services”
- In “services” folder, create a new file named “api.ts” then type code as below:

```
const url = {
  baseUrl: "https://restfulapi.dnd-group.net/api",
  login: "/login",
};

const api = {
  url
};

export default api;
```

```
import axios from "axios";

const url = {
  baseUrl: "https://restfulapi.dnd-group.net/api",
  login: "/login",
};

const instance = axios.create({
  baseURL: url.baseUrl,
  headers: {
    "Content-Type": "application/json",
    Accept: "application/json",
  },
});

const api = {…
};

export default api;
```

```
import axios from "axios";

const url = {…
};

const instance = axios.create({…
});

const api = {
  url,
  instance,
  get: instance.get,
  post: instance.post,
  put: instance.put,
  delete: instance.delete,
  patch: instance.patch,
};

export default api;
```

Create ResponseWrapper Type

```
{
    "errorCode": 0,
    "message": "",
    "data": {
        "name": "New Major",
        "id": 928
    }
}
```

- In services folder, create a new file named "responseWrapper.ts", then type code below

```
type ResponseWrapper<T> = {
  errorCode: number;
  message: string;
  data: T;
};

export default ResponseWrapper;
```

Create userService.ts file

- In “services” folder, create a new file named “userService.ts” then type code as below:

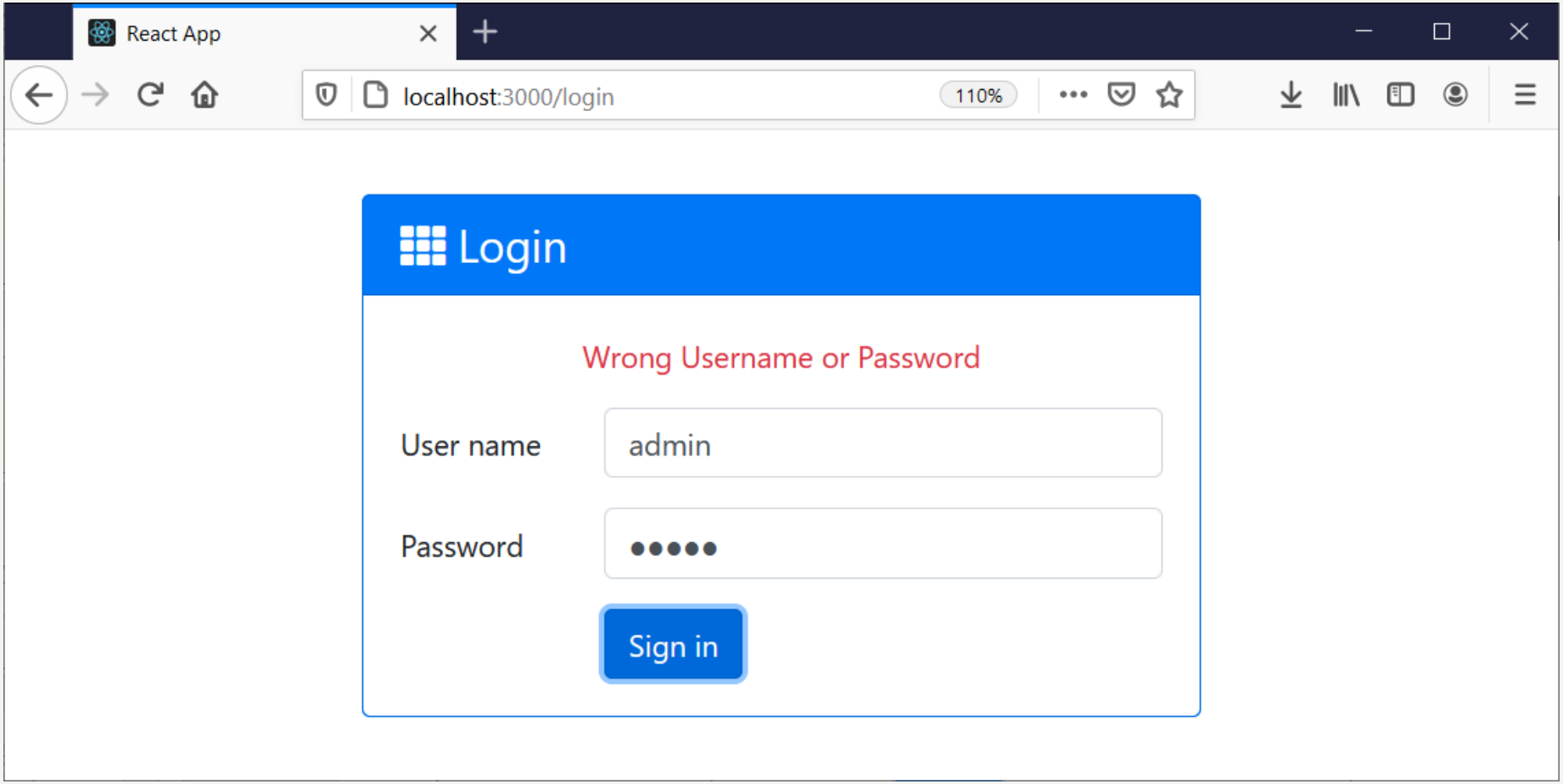
```
import api from "./api";
import ResponseWrapper from "./responseWrapper";

type LoginInfo = {
  id: number;
  username: string;
  fullName: string;
  accessToken: string;
};

const login = (username: string, password: string) => {
  const data = { username, password };
  return api.post<ResponseWrapper<LoginInfo>>(api.url.login, data)
            .then((res) => res.data);
};

const userService = {
  login,
};
export default userService;
```

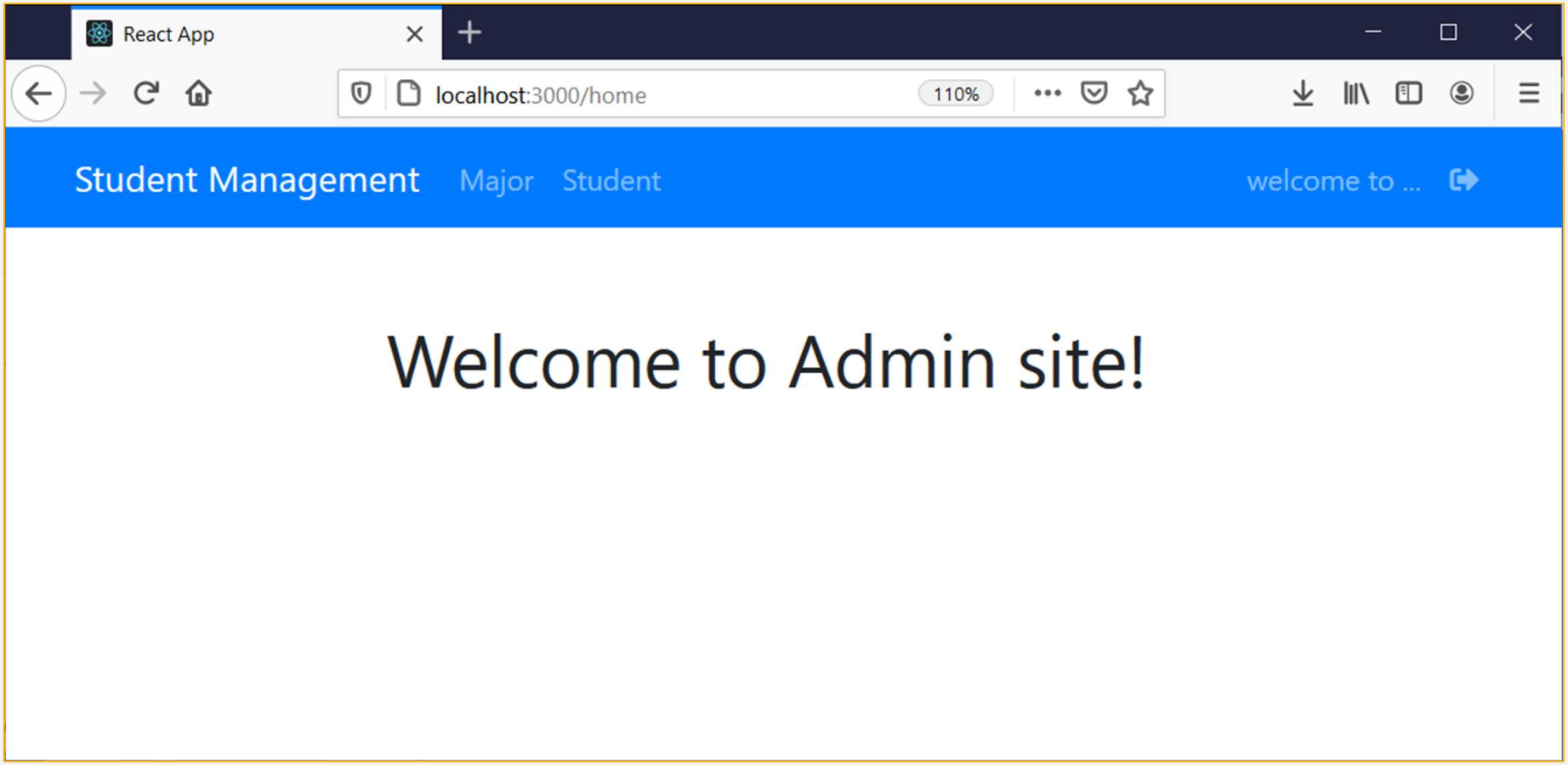
Calling Api from Login.tsx

```tsx
import userService from "./../services/userService";
import { useNavigate } from "react-router-dom";

const Login = () => {
  const navigate = useNavigate();
```

```tsx
const formSubmitHandler = (e: React.FormEvent<HTMLFormElement>) => {
  e.preventDefault();
  const username = usernameRef.current?.value;
  const password = passwordRef.current?.value;
  userService.login(username, password).then((res) => {
    if (res.errorCode === 0) {
      setMessage("");
      navigate("/home");
    } else {
      setMessage(res.message);
    }
  });
};
```

Test


React App
localhost:3000/login
110%
Login
Wrong Username or Password
User name
admin
Password
Sign in


React App
localhost:3000/home
110%
Student Management
Major
Student
welcome to ...
Welcome to Admin site!

Modifying api.ts

```
import axios from "axios";

const url = {
  baseUrl: "https://restfulapi.dnd-group.net/api",
  login: "/login",
  majors: "/majors",
};

const instance = axios.create({ ···
});

const api = { ···
};

export default api;
```

Creating MajorService

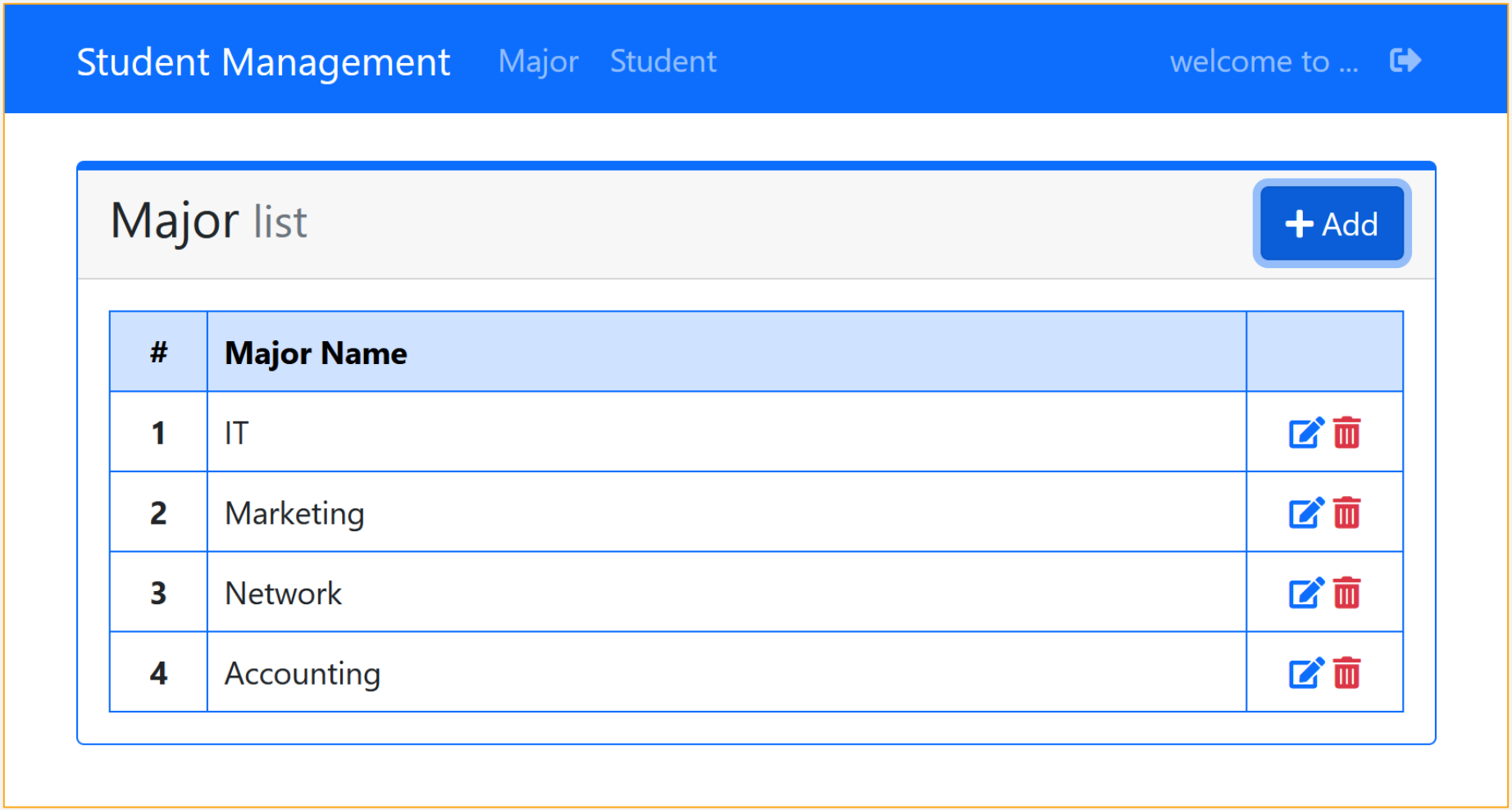
- Create a new file named “majorService.ts” in “services” folder.

```
import api from "./api";
import ResponseWrapper from "./responseWrapper";

export interface IMajor {  id: number;  name: string; }

const list = () => api.get<ResponseWrapper<IMajor[]>>(api.url.majors)
                      .then((res) => res.data);
const get = (id: number) => api.get<ResponseWrapper<IMajor>>(`${api.url.majors}/${id}`)
                                .then((res) => res.data);
const add = (data: IMajor) => api.post<ResponseWrapper<IMajor>>(api.url.majors, data)
                                  .then((res) => res.data);
const update = (id: number, data: IMajor) =>
  api.put<ResponseWrapper<IMajor>>(`${api.url.majors}/${id}`, data)
     .then((res) => res.data);
const remove = (id: number) =>
  api.delete<ResponseWrapper<IMajor>>(`${api.url.majors}/${id}`)
     .then((res) => res.data);

const majorService = { list, get, add, update, delete: remove };
export default majorService;
```

Showing Major list

```tsx
import React, { useEffect, useState } from "react";
import { useNavigate } from "react-router-dom";
import majorService, { IMajor } from "../services/majorService";

const Major = () => {
  const [majors, setMajors] = useState<IMajor[]>([]);
  const navigate = useNavigate();

  const loadData = () => {
    majorService.list().then((res) => setMajors(res.data));
  };

  useEffect(() => {
    loadData();
  }, []);
```

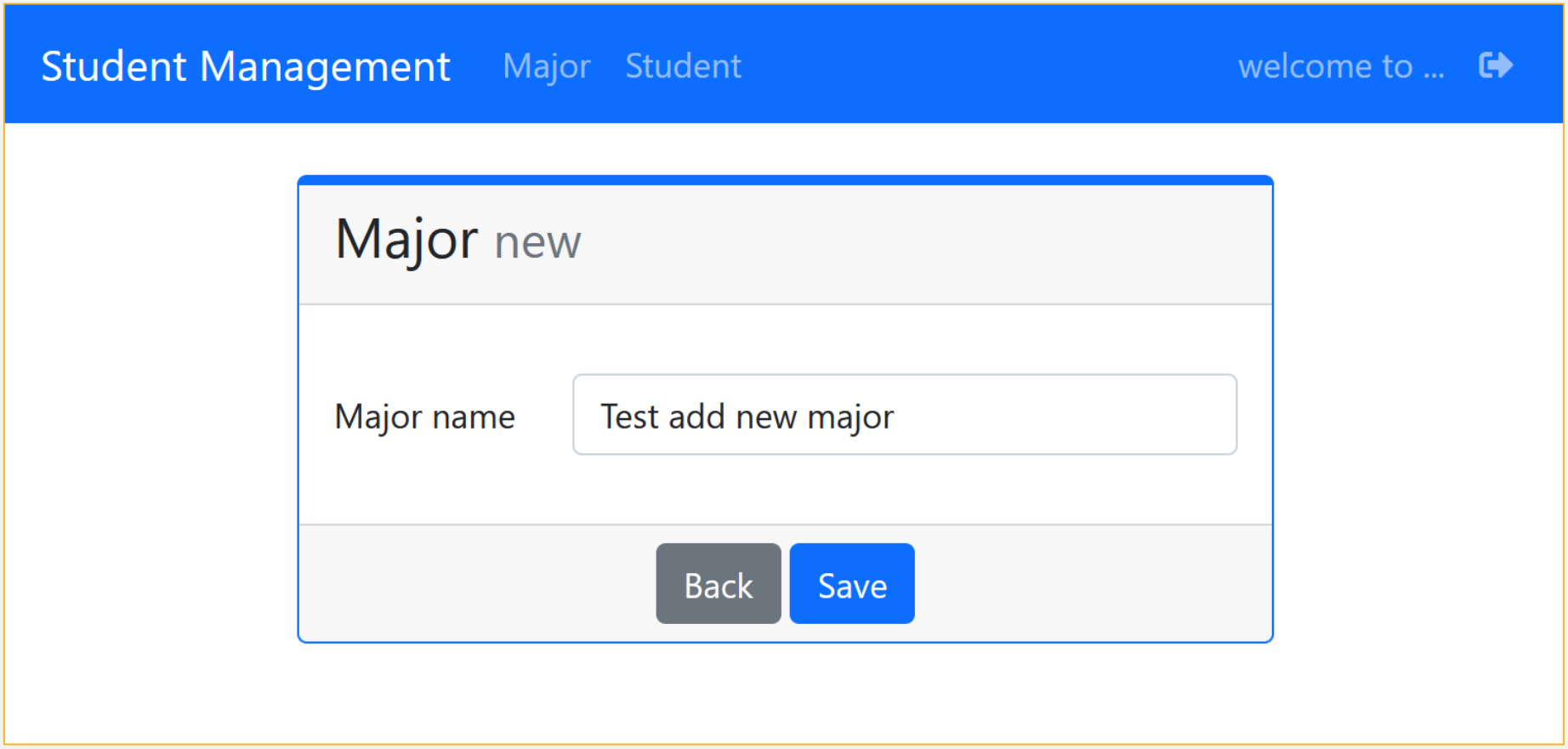
```
<tbody>
  {majors.map((item, index) => (
    <tr key={item.id}>
      <td>{index + 1}</td>
      <td>{item.name}</td>
      <td>
        <a href="/#">
          <i className="bi-pencil-square text-primary" />
        </a>
        <a href="/#">
          <i className="bi-trash text-danger" />
        </a>
      </td>
    </tr>
  ))}
</tbody>
```


Student Management
Major
Student
welcome to ...
Major list
+ Add
# Major Name
1 IT
2 Marketing
3 Network
4 Accounting

Adding a new Major

```
const MajorEdit = () => {
  const { id } = useParams<{ id: string }>();
  const navigate = useNavigate();

  const [major, setMajor] = useState<IMajor>({ id: 0, name: "" });
  const [message, setMessage] = useState("");

  const handleBack = () => {···
  };

  const handleChange = (e: React.FormEvent<HTMLInputElement>) => {···
  };

  const handleSave = () => {
    majorService.add(major).then((res) => {
      if (res.errorCode === 0) navigate("/major");
      else setMessage(res.message);
    });
  };
```

```
<div className="card-body">
  <p className="text-center text-danger">{message}</p>
  <form>
    <Input label="Major name" id="txtMajorName" name="name"
      onChange={handleChange} defaultValue={major.name}
    />
  </form>
</div>
<div className="card-footer text-center">
  <button type="button" className="btn btn-secondary me-1"
    onClick={handleBack}>
    Back
  </button>
  <button type="button" className="btn btn-primary"
    onClick={handleSave}>
    Save
  </button>
</div>
```


Student Management
Major
Student
welcome to ...
Major new
Major name
Test add new major
Back
Save

Showing a Major information

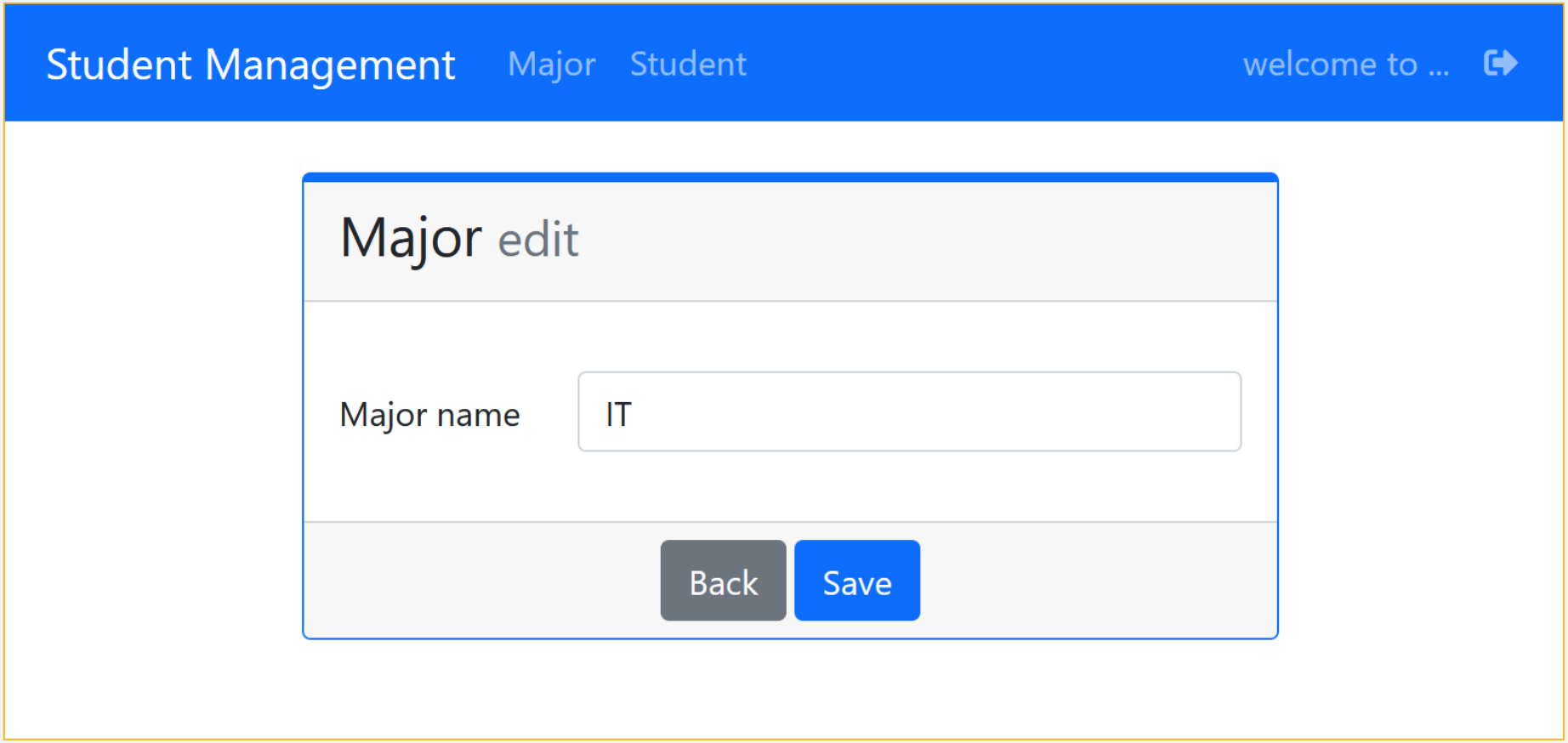
- In Major Component

```
<a href="/#" className="me-1"
  onClick={(e) => showEditPage(e, item.id)}>
  <i className="bi-pencil-square text-primary" />
</a>
```

- In MajorEdit Component

```
useEffect(() => {
  if (!/\d+/.test(id as string)) {
    navigate("/not-found");
  } else if (Number(id) > 0) {
    majorService.get(Number(id)).then(res => setMajor(res.data));
  }
}, [id, navigate]);
```

```
<Input type="text" onChange={handleChange} label="Major name"
  name="name" defaultValue={major.name} />
```


Student Management
Major
Student
welcome to ...
Major edit
Major name
IT
Back
Save

Updating a Major information

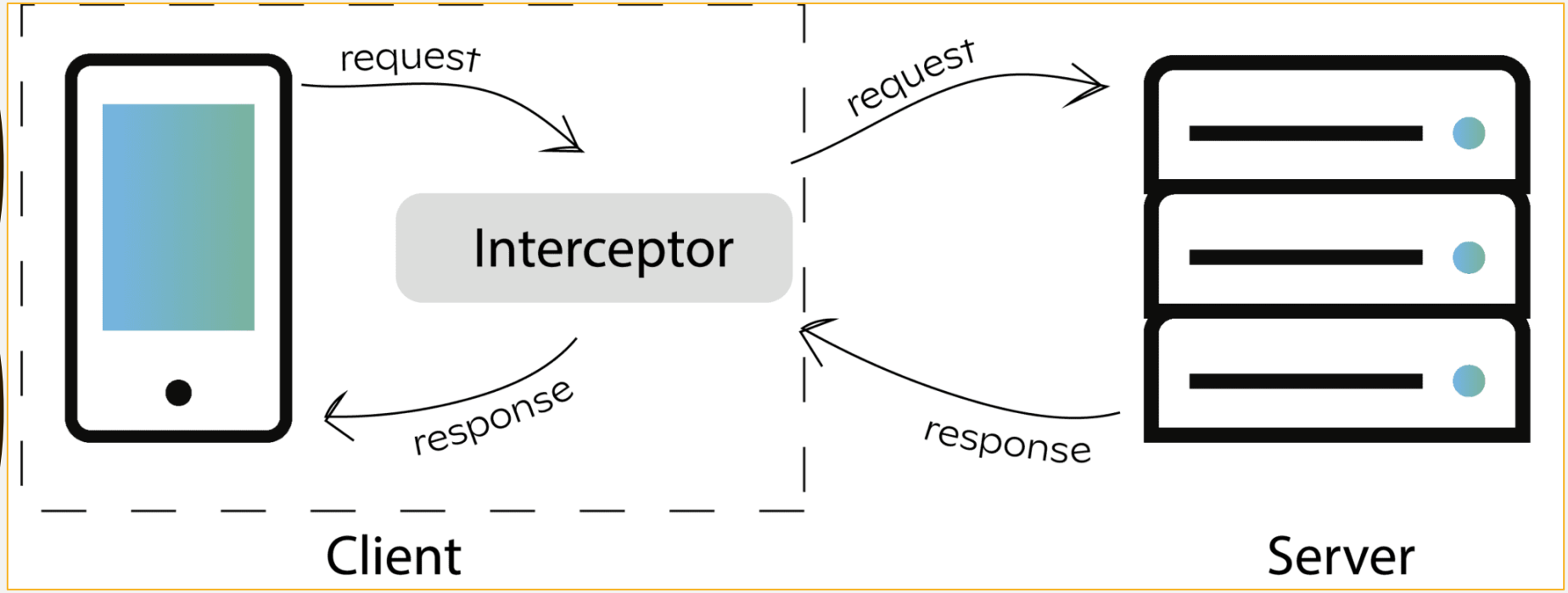
```
const handleSave = () => {
  if (Number(id) === 0) {
    majorService.add(major).then((res) => {
      if (res.errorCode === 0) navigate("/major");
      else setMessage(res.message);
    });
  } else {
    majorService.update(major.id, major).then((res) => {
      if (res.errorCode === 0) navigate("/major");
      else setMessage(res.message);
    });
  }
};
```

Deleting a Major

```
> const showEditPage = (e: any, id: number) => {···
  };

  const handleDelete = (e: React.FormEvent<HTMLAnchorElement>, id: number) => {
    e.preventDefault();
    majorService.delete(id).then((res) => {
      if (res.errorCode === 0) loadData();
    });
  };

  return (
```

```
<td>
  <a href="/#" className="me-1" onClick={(e) => showEditPage(e, item.id)}>
    <i className="bi-pencil-square text-primary" />
  </a>
  <a href="/#" onClick={(e) => handleDelete(e, item.id)}>
    <i className="bi-trash text-danger" />
  </a>
</td>
```

Interceptors

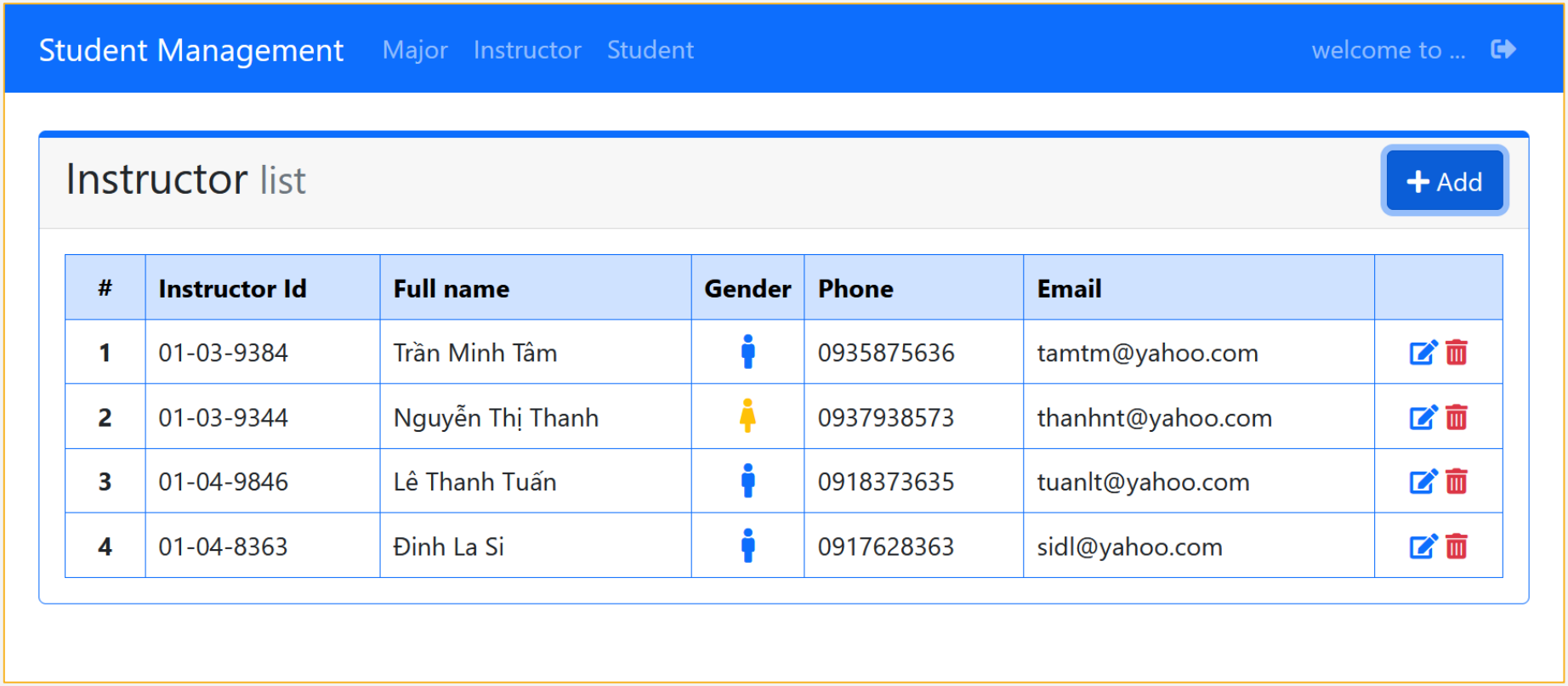


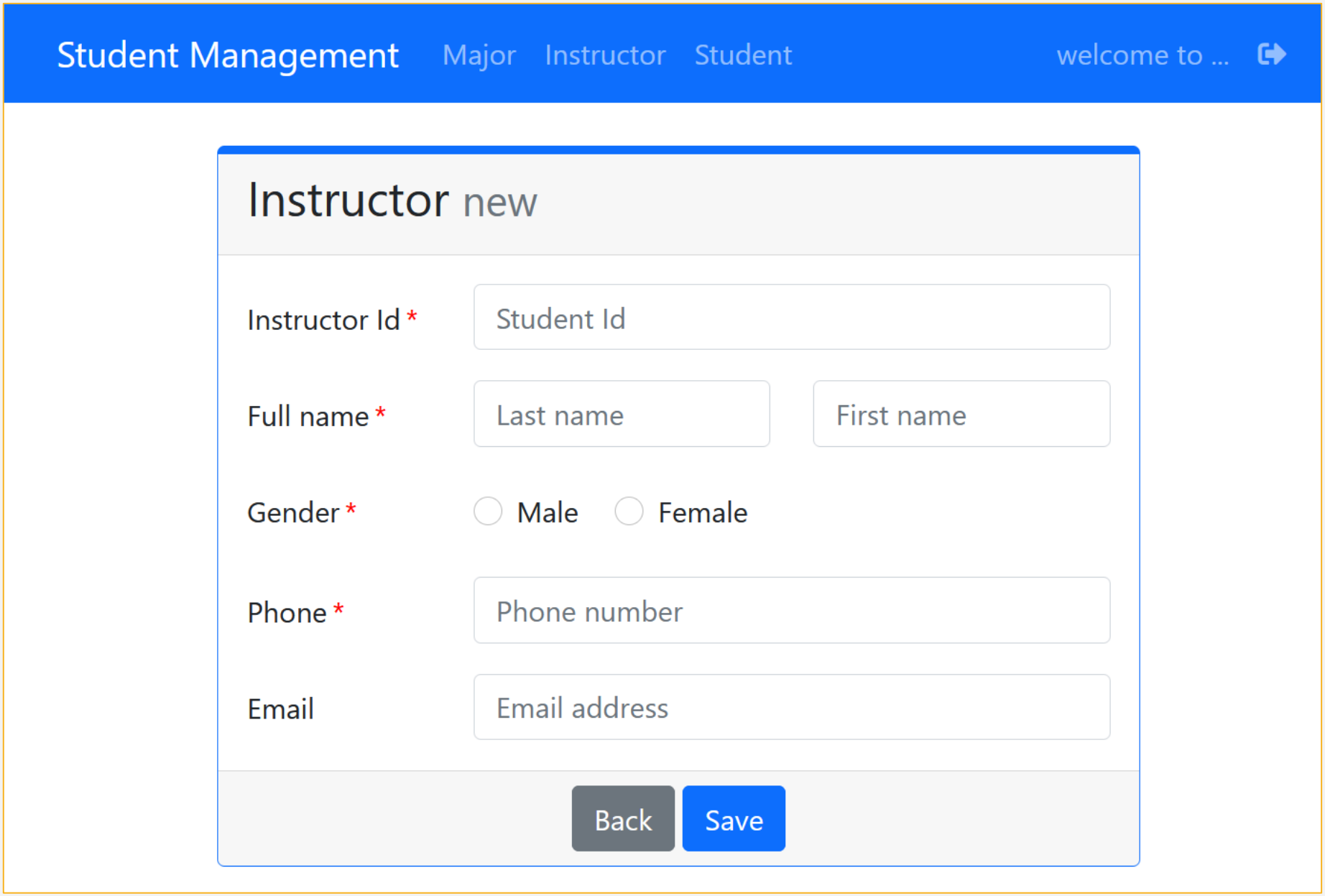
Setting Up Interceptors

```
> const instance = axios.create({ ···
  });
  instance.interceptors.request.use((request) => request);
  instance.interceptors.response.use(
    (response) => response,
    (err) => {
      if (err.code === "ERR_NETWORK") {
        window.location.href = "/network-error";
      } else {
        switch (err.response.status) {
          case 401: window.location.href = "/login"; break;
          case 403: window.location.href = "/no-permission"; break;
          default: break;
        }
      }
      return Promise.reject(err);
    }
  );
```

Exercise

- Implement “Instructor” component using “/instructors” api




Student Management
Major
Instructor
Student
welcome to ...
Instructor new
Instructor Id *
Student Id
Full name *
Last name
First name
Gender *
Male
Female
Phone *
Phone number
Email
Email address
Back
Save

THE END