

## w15-p1: client /testcore to test TestFetchUseCors\_81.js

The screenshot shows the client code in VS Code and the Chrome DevTools network logs. The code in `TestFetchUseCors_81.js` is as follows:

```

const testFetchUseCors_81 = async () => {
  console.log('test');
  const response = await fetch('http://localhost:5000');
  const data1 = await response.json();
  console.log('fetch api data', data1);

  const data2 = await axios.get('http://localhost:5000');
  console.log('axios data2', data2);

  try {
    const currentUser = {
      name: "yangcici",
      email: "123456@gmail.com",
      password: "secret"
    };
    const { data } = await axios.post('http://localhost:5000/api/v1/auth_81/register_81', currentUser);
    console.log("register", data)
  } catch (err) {
    console.log(err);
  }
}

```

The Chrome DevTools network logs show the following requests and responses:

- fetch api data Welcome cici 209410181--**: Response status 200, statusText 'OK', headers: {-}, config: {-}, data: {data: 'Welcome cici 209410181--', status: 200, statusText: 'OK', headers: {-}, config: {-}, -}
- axios data2**: Response status 200, statusText 'OK', headers: {-}, config: {-}, data: {data: 'Welcome cici 209410181--', status: 200, statusText: 'OK', headers: {-}, config: {-}, -}
- register**: Response status 200, statusText 'OK', headers: {-}, config: {-}, data: {data: 'Welcome cici 209410181--', status: 200, statusText: 'OK', headers: {-}, config: {-}, -}
- POST http://localhost:5000/api/v1/auth\_81/register\_81 500**: Internal Server Error (xhr.js:220)
- POST http://localhost:5000/api/v1/auth\_81/register\_81 500**: Internal Server Error (xhr.js:220)

The screenshot shows the server code in VS Code and the MongoDB Atlas interface. The code in `server_81.js` is as follows:

```

import 'express-async-errors'
import morgan from 'morgan'
import cors from 'cors'
import connectDB_81 from './db/connect_81.js';
import authRoutes_81 from './routes/authRoutes_81.js';

//middleware
import notFoundMiddleware_81 from './middleware/not-found_81.js';
import errorHandler_81 from './middleware/error-handler_81.js';

if (process.env.NODE_ENV !== 'production') {
  app.use(morgan('dev'));
}

app.use(cors());
app.use(express.json());

```

The MongoDB Atlas interface shows the following configuration:

- Database**: demo
- Collection**: user\_81
- Query Results**: 1-1 OF 1
- Document**: {\_id: ObjectId("62989bdb35ea9f70575d6e0c"), name: "yangcici", email: "123456@gmail.com", password: "\$2a\$10\$6DmQx2k8ZC0by414UG63JuzvLOXL17cVDdT91Tcn2HoxZREGBML02", lastName: "lastName", location: "my city", \_v: 0}

