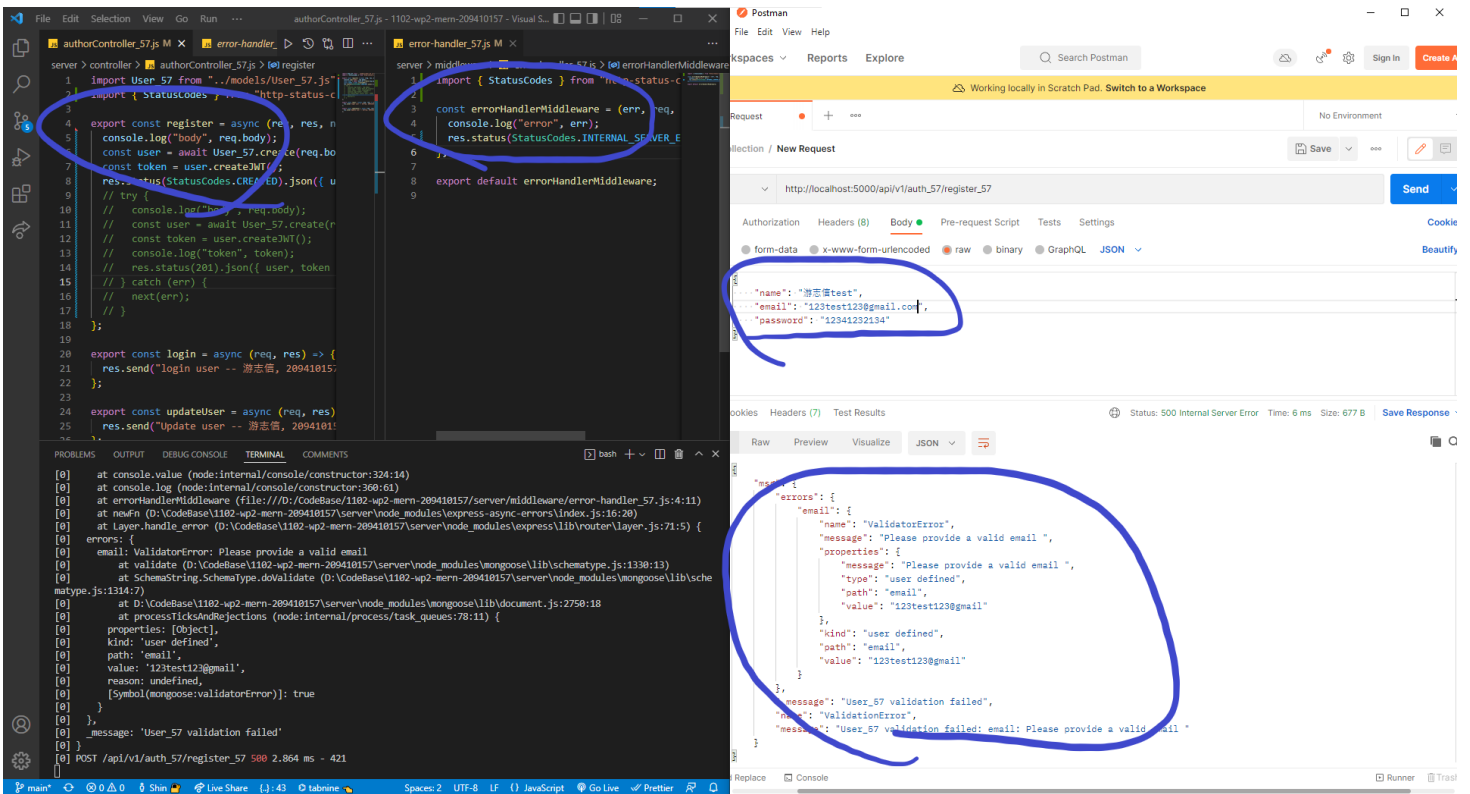


1

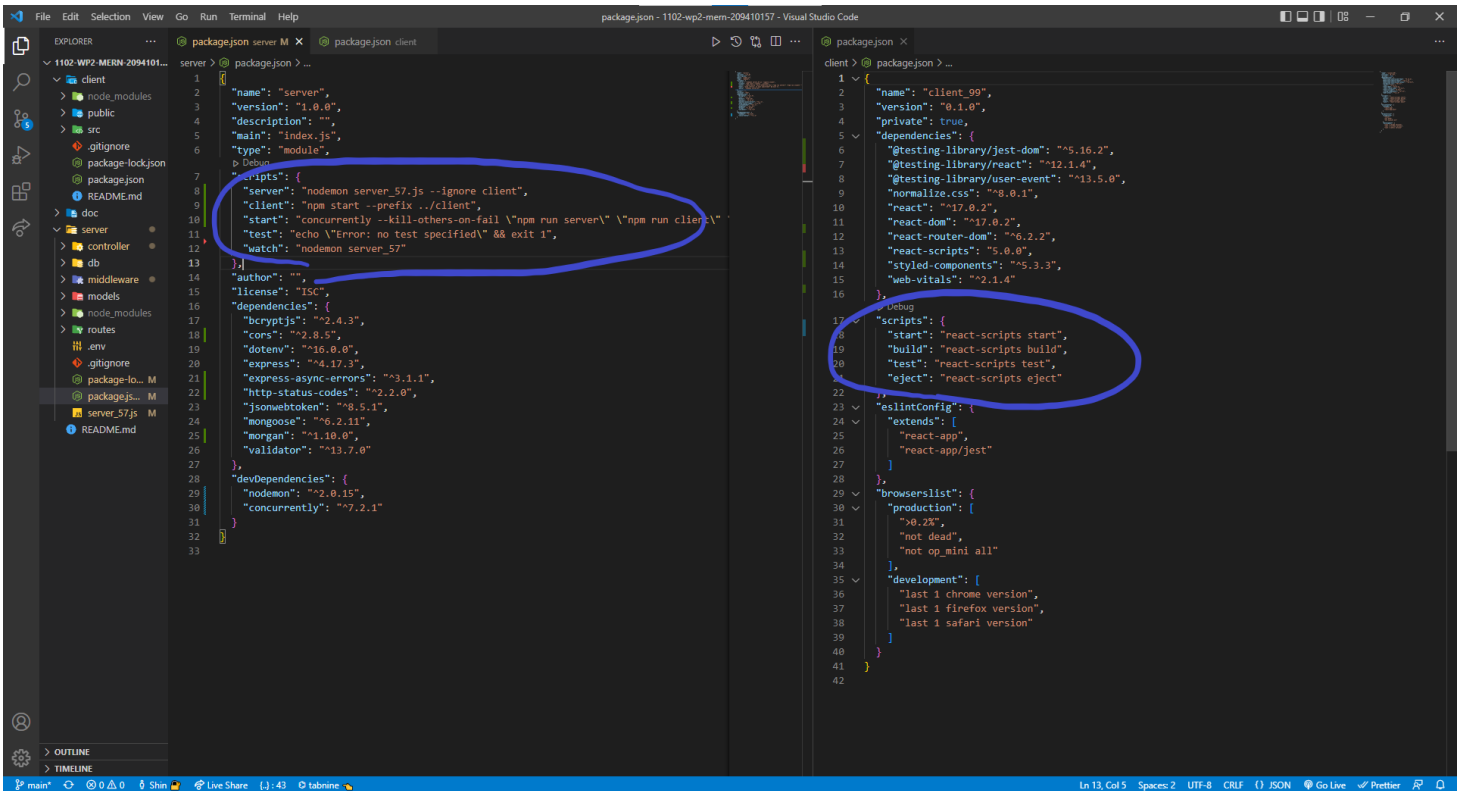
The screenshot displays a development environment with three main components:

- Code Editor:** Shows the `authorController_57.js` file. The `register` function is highlighted with a blue circle. It uses `async/await` to create a user, generate a token, and return a JSON response. The `login` and `updateUser` functions are also visible.
- Terminal:** Displays a detailed error message from Mongoose: `ValidatorError: Please provide a valid email`. The error details include the path (`'email'`), the value (`'test123@gmail'`), and the reason (`undefined`).
- Postman:** Shows a REST client request to `http://localhost:5000/api/v1/auth_57/register_57` with a JSON body: `{ "name": "测试用户test", "email": "test123@gmail", "password": "12341232134" }`. The response is a 500 Internal Server Error with a JSON body: `{ "msg": { "errors": { "email": { "name": "ValidatorError", "message": "Please provide a valid email", "properties": { "message": "Please provide a valid email", "type": "user defined", "path": "email", "value": "test123@gmail" }, "kind": "user defined", "path": "email", "value": "test123@gmail" } }, "_message": "User_57 validation failed", "name": "ValidationError", "message": "User_57 validation failed: email: Please provide a valid email" } }`.

2



3



4

```
client> src> context> appContext_57.js> [9] AppProvider_57> [9] clearAlert
1 import React, { useReducer, useContext } from "react";
2 import reducer_57 from "../reducer_57";
3 import { CLEAR_ALERT, DISPLAY_ALERT } from "../action_57";
4 import axios from "axios";
5
6 const initialState = {
7   isLoading: false,
8   showAlert: false,
9   alertText: "",
10  alertType: ""
11 };
12
13 const AppContext_57 = React.createContext();
14
15 const AppProvider_57 = ({ children }) => {
16   const [state, dispatch] = useReducer(reducer_57, initialState);
17
18   const displayAlert = () => {
19     dispatch({ type: DISPLAY_ALERT });
20     clearAlert();
21   };
22
23   const clearAlert = () => {
24     setTimeout(() => {
25       dispatch({ type: CLEAR_ALERT });
26     }, 3000);
27   };
28
29   const registerUser = async ({ currentUser, endPoint, alertText }) => {
30     console.log(endPoint);
31     console.log(currentUser);
32     await axios.post(
33       'http://localhost:5000/api/v1/auth 57/register 57',
34       currentUser
35     );
36   };
37
38   return (
39     <AppContext_57.Provider
40       value={{ ...state, displayAlert, clearAlert, registerUser }}
41     >
42       {children}
43     </AppContext_57.Provider>
44   );
45 };
46
47 const useAppContext = () => {
48   return useContext(AppContext_57);
49 };
```

