

GithubContext.js

```
import { createContext, useReducer } from "react"; // import
useReducer hook
import githubReducer from "../GithubReducer"; // import our function

const GithubContext = createContext();

const GITHUB_URL = process.env.REACT_APP_GITHUB_URL;
const GITHUB_TOKEN = process.env.REACT_APP_GITHUB_TOKEN;

export const GithubProvider = ({ children }) => {
  const initialState = { // We set users and loading as default at
    here.
    users: [],
    loading: false
  }

  const [state, dispatch] = useReducer(githubReducer, initialState)

  const fetchUsers = async () => {
    const response = await fetch(`${GITHUB_URL}/users`, {
      headers: {
        Authorization: `token ${GITHUB_TOKEN}`,
      },
    });
    const data = await response.json();

    dispatch({ // We define type for use at githubReducer, data
      comes from const data above.
      type: "GET_USERS",
      payload: data,
    })
  }

  return <GithubContext.Provider value={{
    users: state.users, //we change here cause we use reducer.Now
    These are coming from state we define above ([state,dispatch])
  }} />
```

```

        loading: state.loading,
        fetchUsers
    }}>
    {children}
  </GithubContext.Provider>
}

export default GithubContext

```

GithubReducer.js

```

const githubReducer = (state, action) => { /Takes to param
  switch (action.type) { /if action.type is true
    case "GET_USERS": / Case GET_USERS run,we defined it at
dispatch
      return {
        ...state,
        users: action.payload, / .payload could be something else
but most used syntax is payload, when payload is run.(payload: data
runs to.)-- action comes from action.type
        loading: false
      }
    default:
      return state
  }
}

export default githubReducer

```