

# XUI development tip

When customizing the XUI which involves changing the JavaScript code, it is recommended to setup a development environment as described in the documentation here - <https://backstage.forgerock.com/docs/am/6.5/ui-customization-guide/#xui-testing-deploying> and <https://backstage.forgerock.com/docs/am/6.5/ui-customization-guide/#xui-dev-server>.

The second article talks about how to setup a XUI only development server which connects to a separate AM instance. This eliminates the need for building and deploying XUI everytime a change is made. While that article assumes that the XUI dev server and AM are running on the same host, this articles discusses the steps where the XUI development server can be the local machine (laptop) and the AM could be a remote host.

The process is quite straightforward, we just configure the webpack development server (already part of XUI) to proxy all non-XUI requests to the remote host (instead of the local host, which is the default).

We are making the following assumptions:

1. The AM server is running on a remote host, `openam.example.com` as `http://openam.example.com:8080/openam`
2. The XUI development server/environment is running on your laptop, and you have configured `xui.example.com` to point to `127.0.0.1` in your local `/etc/hosts` (make sure that the domain name you use for you local host is the same as the AM server's - so the `iPlanetDirectoryPro` cookie comes across).
3. Make sure that the application context of AM, mentioned in the lines below (`/openam/XUI` by default), reflects what you have in your environment:

```
.
.
proxy: {
  "!/openam/XUI": {
    changeOrigin: true,
    router: (request) => {
      //const requestUrl = new URL(`${request.protocol}://${request.get("host")}`);
      const requestUrl = new URL(`http://openam.example.com`);
      requestUrl.port = PROXY_TARGET_PORT;
      return requestUrl;
    },
    target: `http://localhost:${PROXY_TARGET_PORT}`
  }
},
publicPath: "/openam/XUI/"
.
.
```

## Step-by-step guide

1. Follow the instructions in <https://backstage.forgerock.com/docs/am/6.5/ui-customization-guide/#xui-download-project> to clone and setup a local development environment on your laptop
2. Open `<project dir>/openam-ui/openam-ui-ria/config/webpack/configurations/development.js` in a text editor
3. Change the line which reads

```
const requestUrl = new URL(`${request.protocol}://${request.get("host")}`);
```

to

```
const requestUrl = new URL(`http://openam.example.com`);
```

If the remote AM instance is not running on port 8080, change

```
requestUrl.port = PROXY_TARGET_PORT;
```

to

```
requestUrl.port = 80;
```

(for example when remote AM is running on port 80).

4. Start/restart the development server

```
yarn start
```

You should be able to access the locally running XUI in webpack development server by going to `http://xui.example.com:8080/openam`. Any changes you make to the code will now be instantly compiled and be reflected on the above address.

## Related articles

- [XUI development tip](#)

