

Deploying the theme

Blueriq has a standard Material Theme packaged but when creating your own theme this must be deployed somehow. You need a web server for this that conforms to these requirements:

1. The web server should be able to serve static content
2. The web server should be able to proxy API requests to the runtime

By proxying the runtime requests, the theme and the backend share the same origin, so XHR-requests from theme to runtime are allowed by the default browser security policy.

Suitable web servers include, but are not limited to nginx, Apache httpd and IIS.

Yarn build

The Material Kickstarter comes with 2 build scripts and both use a different environment.ts file which can be configured if needed. Configuring in this case means changing the baseUrl.

1. Change the `environment.runtime.ts` baseUrl to `/server`
2. Run `yarn build:runtime` (which uses the environment.runtime.ts's configuration)
3. When finished: Copy the content of /dist to a application server you prefer (for nginx that would be: /html)

nginx

For getting started with nginx please take a look at their documentation: <https://www.nginx.com/resources/wiki/>

The following configuration is the minimum needed to get the theme up and running. You could need adjustments to fit your own specific setup.

nginx.conf

```
error_log logs/error.log;

events {
    worker_connections 1024;
}

http {
    upstream revProxyRuntime {
        # 1: Replace <host>:<port>, example: my.runtime.local:8080
        server <host>:<port>;
    }

    server {
        include ../conf/mime.types;
        listen 1337;

        # Instruct the browser to always verify that its cache is up-to-date
        add_header Cache-Control 'max-age=0, must-revalidate';

        location / {
            root html;
            index index.html index.htm;
            try_files $uri $uri/ /index.html =404;
        }

        # 2: /server matches with what was configured in the environment.ts baseUrl
        location /server/ {
            proxy_set_header Host $host;
            proxy_set_header X-Forwarded-Host $host:$server_port;
            proxy_set_header X-Forwarded-Server $host;
            proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
            proxy_set_header X-Forwarded-Proto $scheme;
            proxy_set_header X-Forwarded-Port $server_port;

            # 3: This uses the upstream (step 1): So this becomes: http://<host>:<port>/<context>
            /server/$and_the_rest_of_it;
            # Change the <context> part to the context root of the runtime. This is 'Runtime' if
            you installed Blueriq with the installer.
            proxy_pass http://revProxyRuntime/<context>$request_uri;

            proxy_cookie_path /<context> /;
        }
    }
}
```

External Theme

When your custom theme is configured as an external theme in the Development Tools Component, you should add an additional location block for /runtime calls. Without doing so the development toolbar is unable to retrieve session information from the runtime when you start a project using the your custom theme as the external theme. More information about configuring external themes can be found [here](#).

nginx.conf

```
error_log logs/error.log;

events {
    worker_connections 1024;
}

http {
    upstream revProxyRuntime {
        # 1: Replace <host>:<port>, example: my.runtime.local:8080
        ...
    }

    server {
        include ../conf/mime.types;
        listen 1337;

        # Instruct the browser to always verify that its cache is up-to-date
        add_header Cache-Control 'max-age=0, must-revalidate';

        location / {
            ...
        }

        # 2: /server matches with what was configured in the environment.ts baseUrl
        location /server/ {
            ...
        }

        # 3: /runtime matches Runtime calls which are made by the development toolbar
        location /runtime/ {
            proxy_set_header Host $host;
            proxy_set_header X-Forwarded-Host $host:$server_port;
            proxy_set_header X-Forwarded-Server $host;
            proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
            proxy_set_header X-Forwarded-Proto $scheme;
            proxy_set_header X-Forwarded-Port $server_port;

            proxy_pass http://revProxyRuntime$request_uri;

            proxy_cookie_path /Runtime /;
        }
    }
}
```