

Node.js: create a QR code in ExpressJS

In this article we will see how to create a QR code in ExpressJS.

We install the required NPM package.

```
npm install qrcode --save
```

This module can create PNG files both as data URLs and as actual files. So let's define a utility function for file management.

```
'use strict';

const fs = require('fs');
const util = require('util');

const exists = util.promisify(fs.access);

module.exports = {
  fileExists(path) {
    return new Promise((resolve, reject) => {
      exists(path, fs.F_OK).then(ok => {
        resolve(true);
      }).catch(err => {
        resolve(false);
      });
    });
  }
};
```

```
    }  
};
```

We then define a route with conditional logic: if the GET request is made with AJAX, a data URL is generated, otherwise it is served the PNG file created.

```
'use strict';  
  
const fs = require('fs');  
const path = require('path');  
const express = require('express');  
const QRCode = require('qrcode');  
const utils = require('./utils');  
const app = express();  
const port = process.env.PORT || 3000;  
  
app.disable('x-powered-by');  
app.use('/public',  
express.static(path.join(__dirname, 'public')));  
  
app.get('/qrcode', async (req, res) => {  
  try {  
  
    const qrCodeText = 'https://site.tld';  
  
    if(req.xhr) {  
      const url = await  
QRCode.toDataURL(qrCodeText);  
      res.json({url});  
    } else {  
      const src = './public/images/qrcode.png';
```

```

        const exists = await
utils.fileExists(src);
        if(!exists) {
            const stream =
fs.createWriteStream(src);
            const code = await
QRCode.toFileStream(stream, qrCodeText);
        }
        res.sendFile(src);
    }
} catch(err) {
    res.json(err);
}
});

app.listen(port);

```

A typical use of the AJAX mode is when the QR code is displayed asynchronously as in the following example.

```

"use strict";

$(function() {
    $.get( "/qrcode", function( response ) {
        $( "#qrcode-image" ).attr( "src",
response.url ).show();
    });
});

```