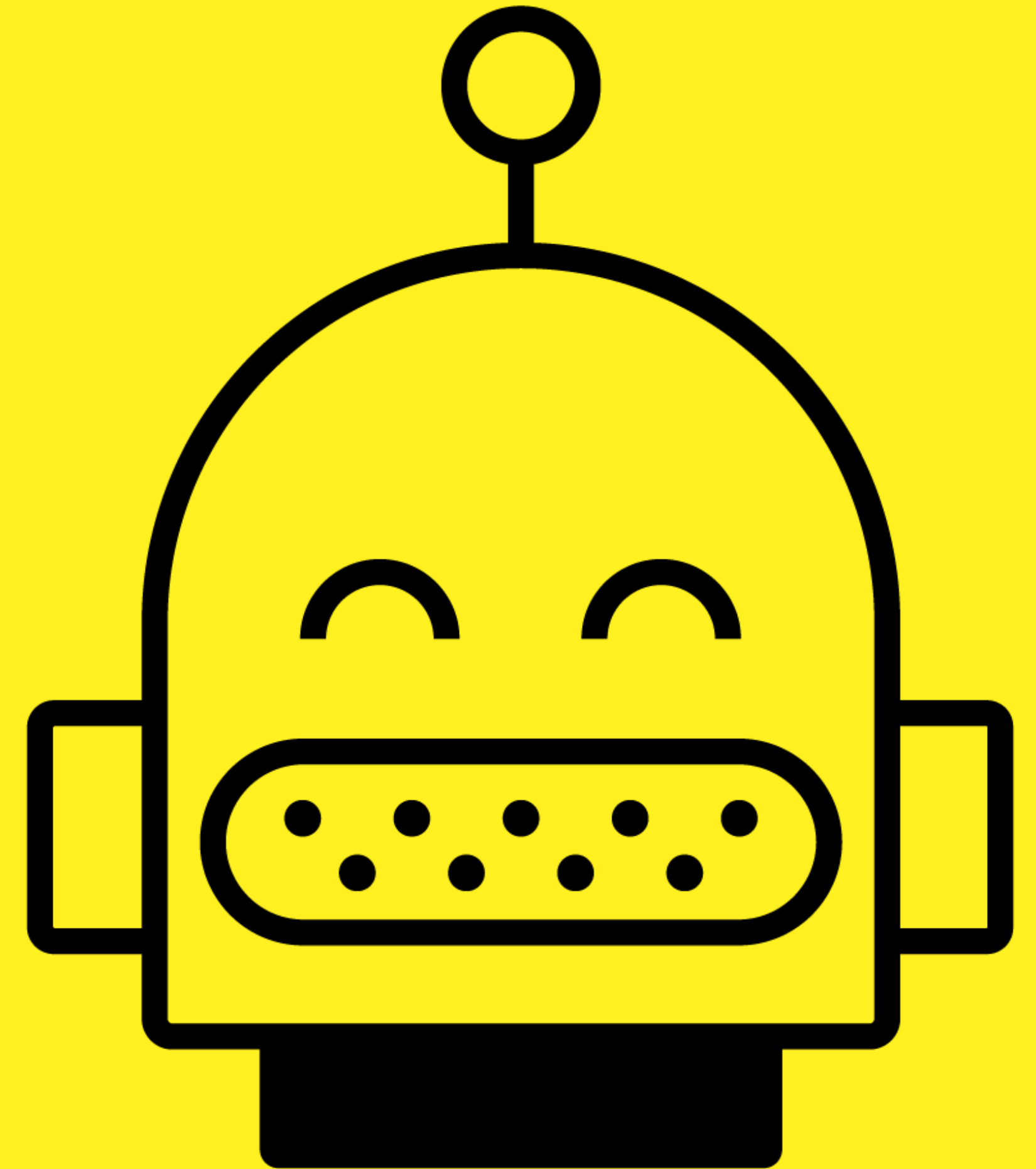




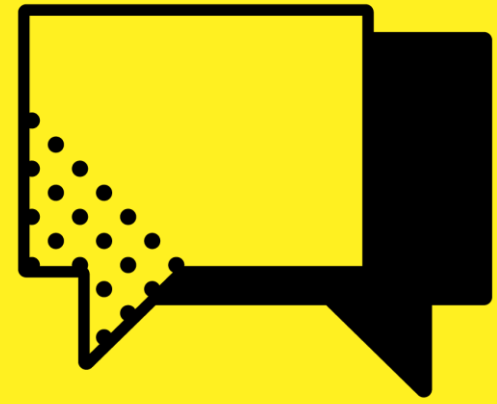
# Project Tech

## Les 7.1

Performance & build tools



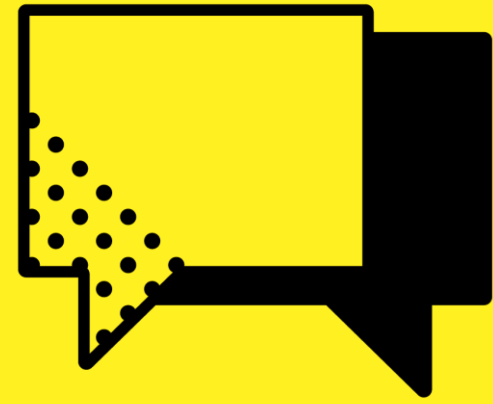
Creating Tomorrow



# Performance

Een site met 'goede performance':

- Laad snel (gebruikers klikken weg als er na 2 sec geen pagina staat)
- Bespaart energie en CO<sub>2</sub> uitstoot
- Komt hoger in Google search
- Spaart de databundel van je gebruikers



# Optimaliseer afbeeldingen



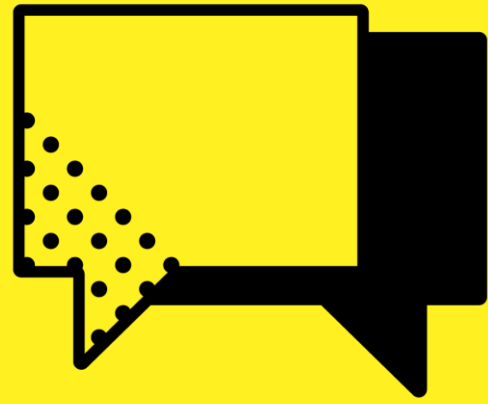
JPG: 4600 x 3450 pixels  
5 MB = 5000 kB



JPG: 500 x 375 pixels  
90 kB



JPG kwaliteit 10%  
40 kB



# Minify (frontend) JS / CSS

```
/*!
 * jQuery JavaScript Library v3.6.0
 * https://jquery.com/
 *
 * Includes Sizzle.js
 * https://sizzlejs.com/
 *
 * Copyright OpenJS Foundation and other contributors
 * Released under the MIT license
 * https://jquery.org/license
 *
 * Date: 2021-03-02T17:08Z
 */
( function( global, factory ) {

    "use strict";

    if ( typeof module === "object" && typeof module.exports === "object" ) {

        // For CommonJS and CommonJS-like environments where a proper `window`
        // is present, execute the factory and get jQuery.
        // For environments that do not have a `window` with a `document`
        // (such as Node.js), expose a factory as module.exports.
        // This accentuates the need for the creation of a real `window`.
        // e.g. var jQuery = require("jquery")(window);
        // See ticket #14549 for more info.
        module.exports = global.document ?
            factory( global, true ) :
            function( w ) {
                if ( !w.document ) {
                    throw new Error( "jQuery requires a window with a document" );
                }
                return factory( w );
            };
    } else {
        factory( global );
    }

} );
```

jQuery library  
282 kB

```
/*! jQuery v3.6.0 | (c) OpenJS Foundation and other contributors | jquery.org/license */
!function(e,t){"use strict";"object"==typeof module&&"object"==typeof module.exports?
module.exports=e.document?t(e,!0):function(e){if(!e.document)throw new Error("jQuery
requires a window with a document");return t(e)}:t(e)}("undefined"!=typeof window?
window:this,function(C,e){"use strict";var t=
[],r=Object.getPrototypeOf,s=t.slice,g=t.flat?function(e){return
t.flat.call(e)}:function(e){return t.concat.apply([],e)},u=t.push,i=t.indexOf,n=
{},o=n.toString,v=n.hasOwnProperty,a=v.toString,l=a.call(Object),y={},m=function(e)
{return"function"==typeof e&&"number"!=typeof e.nodeType&&"function"!=typeof
e.item},x=function(e){return null!=e&&e===e.window},E=C.document,c=
{type:!0,src:!0,nonce:!0,noModule:!0};function b(e,t,n){var r,i,o=
(n=n||E).createElement("script");if(o.text=e,t)for(r in c){i=t[r]||t.getAttribute&&t.getAttribute(r)}&&o.setAttribute(r,i);n.head.appendChild(o).
parentNode.removeChild(o)}function w(e){return null==e?"":!0:"object"==typeof
e||"function"==typeof e?n[o.call(e)]||"object":typeof e}var f="3.6.0",S=function(e,t)
{return new S.fn.init(e,t)};function p(e){var t=!!e&&"length"in
e&&e.length,n=w(e);return!m(e)&&!x(e)&&("array"===n||0===t||"number"==typeof t&&0<t&&t-
1 in e)}S.fn=S.prototype={jquery:f,constructor:S,length:0,toArray:function(){return
s.call(this)},get:function(e){return null==e?s.call(this):e<0?
this[e+this.length]:this[e]},pushStack:function(e){var
t=S.merge(this.constructor(),e);return t.prevObject=this,t},each:function(e){return
S.each(this,e)},map:function(n){return this.pushStack(S.map(this,function(e,t){return
n.call(e,t,e)}))},slice:function(){return
this.pushStack(s.apply(this,arguments))},first:function(){return
this.eq(0)},last:function(){return this.eq(-1)},even:function(){return
this.pushStack(S.grep(this,function(e,t){return(t+1)%2}))},odd:function(){return
this.pushStack(S.grep(this,function(e,t){return t%2}))},eq:function(e){var
t=this.length,n=+e+(e<0?t:0);return this.pushStack(0<=n&&n<t?[this[n]]:
[])},end:function(){return
this.prevObject||this.constructor()},push:u,sort:t.sort,splice:t.splice},S.extend=S.fn.
extend=function(){var e,t,n,r,i,o,a=arguments[0]||
{},s=1,u=arguments.length,l=!1;for("boolean"==typeof a&&(l=a,a=arguments[s]||
{}
```

jQuery minified  
88 kB



# Lazy loading

- Resources (afbeeldingen, scripts, modules) downloaden op het moment dat ze nodig zijn
- Het verbetert de laadtijd van de pagina
- Efficiënter gebruik van bronnen en energie



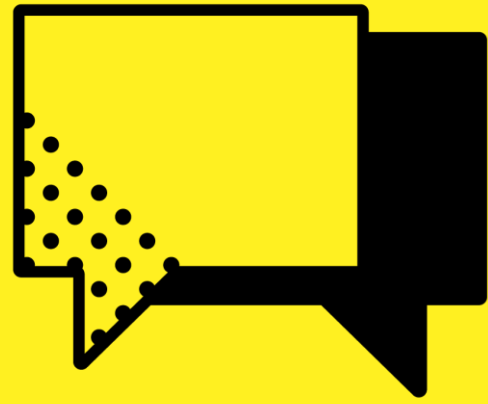


# Lazy loading

Voorbeeld  
module

```
// Normale import van een module  
import Module from './module';
```

```
// Lazy loading van een module met dynamic import  
document.getElementById('button')  
  .addEventListener('click', async () => {  
    const Module = await import('./module');  
  });
```

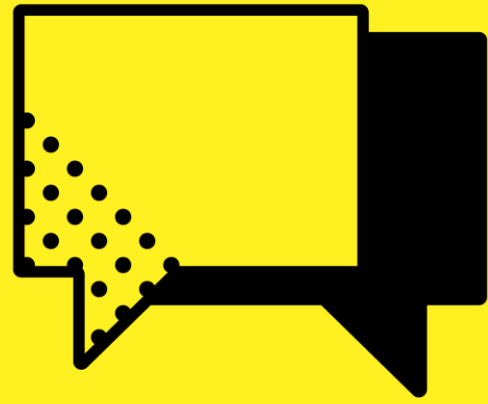


# Lazy loading

Voorbeeld  
afbeeldingen

```
  
  

```



# Lazy loading

Voorbeeld  
afbeeldingen

```
document.addEventListener("DOMContentLoaded", function() {  
  const images = document.querySelectorAll('.lazy-load');  
  
  const imageObserver = new IntersectionObserver((entries, observer) => {  
    entries.forEach(entry => {  
      if (entry.isIntersecting) {  
        const image = entry.target;  
        image.src = image.dataset.src;  
        observer.unobserve(image);  
      }  
    });  
  }, { rootMargin: "0px 0px 50px 0px" });  
  
  images.forEach(image => {  
    imageObserver.observe(image);  
  });  
});
```



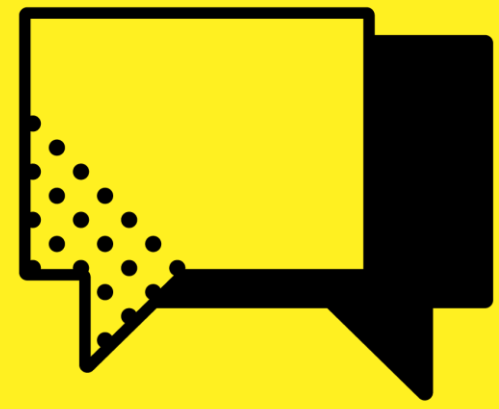
Lesopdracht

# Performance



- Analyseer een paar favoriete sites met Google Pagespeed Insights en met de Website Carbon Calculator
- Welke tips kun je in je eigen project toepassen?



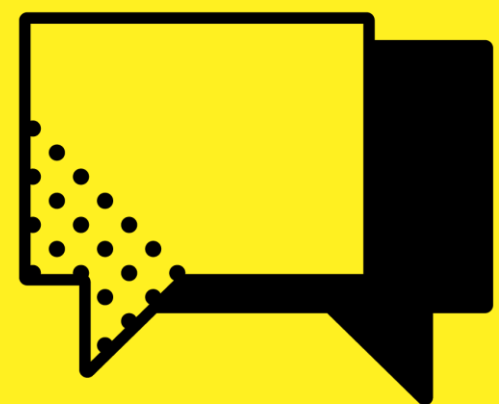


# Build tooling

Om te komen van werkende code tot goede, production-ready code, wil je misschien:

- Je code linten?
- Je plaatjes optimaliseren?
- Je (frontend) JS en CSS minify-en, maar wel het origineel houden om verder aan te bouwen?
- Al je losse styling info samenvoegen tot één CSS?

Dat kan elke keer met de hand, maar developers houden van automatiseren ;)

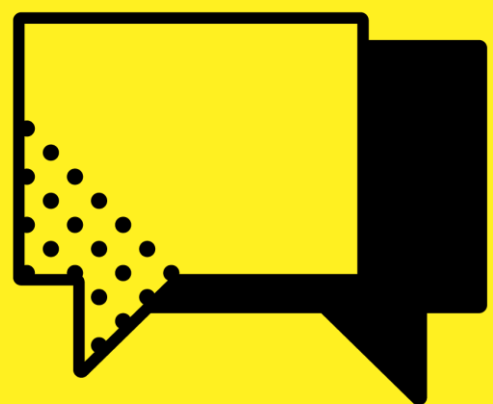


# NPM scripts

Remember:  
package.json?  
Daarin kun je  
scripts aanmaken



```
{  
  "name": "database_api",  
  "version": "1.0.0",  
  "description": "",  
  "main": "server.js",  
  "scripts": {  
    "test": "echo \"Error: no test specified\" && exit 1"  
  },  
  "author": "Ivo Nijhuis",  
  "license": "GPL-3.0-or-later",  
  "dependencies": {  
    "cors": "^2.8.5",  
    "dotenv": "^16.3.1",  
    "express": "^4.18.2",  
    "mongodb": "^6.3.0"  
  }  
}
```



# NPM scripts

```
"scripts": {  
  "start": "node server.js",  
  "trialrun": "nodemon server.js",  
  "deploy": "eslint server.js && node server.js"  
}
```

```
Windows PowerShell  
PS D:\Surfdrive\Project Tech\nieuwe stijl ontwikkelen\database_api> npm run deploy  
  
> database_api@1.0.0 deploy  
> eslint server.js && node server.js  
  
D:\Surfdrive\Project Tech\nieuwe stijl ontwikkelen\database_api\server.js  
443:20  error  'next' is defined but never used  no-unused-vars  
449:25  error  'next' is defined but never used  no-unused-vars  
  
✖2 problems (2 errors, 0 warnings)  
  
PS D:\Surfdrive\Project Tech\nieuwe stijl ontwikkelen\database_api> |
```

Lesopdracht

# Build tooling



Brainstorm met je team:

- Welke taken kunnen jullie met build-tooling automatiseren?







Einde van de les

**Tot de volgende keer!**

