

# Accessibility.

DEVbabbl #3

# Goei'n avond!

[tristandubbeld.nl](http://tristandubbeld.nl)



CodeSandbox



.codesandbox  
project.json  
workspace.json

public  
scripts  
src

actions  
components  
css  
data  
element  
excalidraw-app  
hooks  
locales  
packages  
renderer

/\* index.js  
/\* renderElement.ts  
/\* renderScene.ts  
/\* roundReact.ts

scene  
/\* align.ts  
/\* analytics.ts  
/\* appState.ts  
/\* bug-issue-template.ts

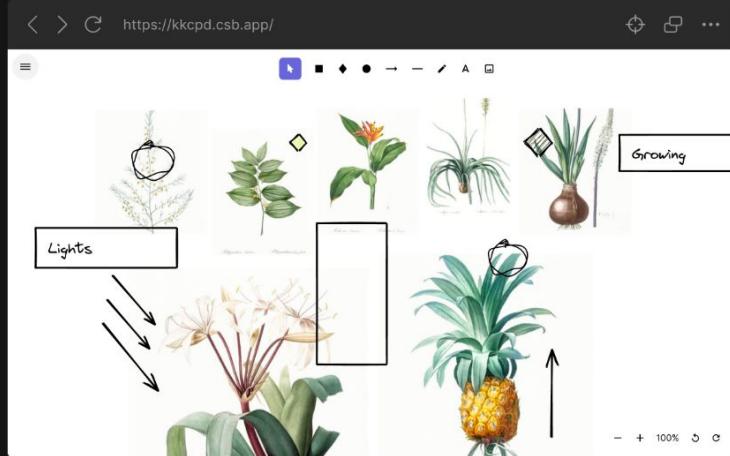
[Open in VS Code](#)

index.ts renderScene.js

```

56 const strokeRectWithRotation = (
57   context: CanvasRenderingContext2D,
58   x: number,
59   y: number,
60   width: number,
61   height: number,
62   cx: number,
63   cy: number,
64   angle: number,
65   fill: boolean = false,
66 ) => {
67   context.save();
68   context.translate(cx, cy);
69   context.rotate(angle);
70   if (fill) {
71     context.fillRect(x - cx, y - cy, width, height);
72   }
73   context.strokeRect(x - cx, y - cy, width, height);
74   context.restore();
75 };
76
77 const strokeDiamondWithRotation = (
78   context: CanvasRenderingContext2D,
79   width: number,
80   height: number,
81   cx: number,
82   cy: number,
83   angle: number,
84 ) => {
85   context.save();

```



> Log - Application

Compiled successfully!

You can now view excalidraw in the browser.

Local: http://localhost:3000  
On Your Network: http://192.168.43.1:3000

Note that the development build is not optimized.  
To create a production build, use `yarn build`.

# Accessibility.



# Waarom?

We're all just temporarily abled.

**We're all just temporarily abled.**

We're all just temporarily abled.

**>40 slechter zicht.**

**>45 cognitief slechter.**

**25% van NL heeft een beperking.**

# Vergrijzing.



**Eigen belang.**

**Arm breken.**

**Solve for one, extend to many.**

**Voor de business.**



# EU accessibility act 2025.



# Boete.



**Als webshop.**

**Klanten wegjagen.**

# Tjarda.





imgPlay

**Realiteit is minder grappig.**

**Hoi, ik ben Tjarda  
en ik ben bijna blind...**

**...en dit is hoe ik jouw ontoegankelijke  
website gebruik.**



• • •

**Webshops zijn niet makkelijk  
voor iedereen.**

In 2022 werden er **50.829.406** verschillende toegankelijkheid fouten gedetecteerd op 1 miljoen gecontroleerde homepages. Dat zijn er **gemiddeld 50,8 per pagina.**

**83,9% tekst contrast.**

**55,4% afbeeldingen.**

**50,1% lege links.**

**46,1% inputs zonder label.**

**Klacht.**

**“Hoi ik ben Tjarda  
en ik ben bijna blind...”**

**“Met de site van Eneco is voor mij slecht te werken. Omdat ik blind ben, dien ik met spraaksoftware te werken op de pc. Een aantal sites zijn tegenwoordig goed bereikbaar voor mij. Maar de kwaliteit van de site van Eneco is zodanig, dat er door mij heel moeilijk mee te werken is. Graag een reactie.”**

A blurry, out-of-focus photograph of a person sitting at a desk. The person is wearing a light-colored shirt and appears to be looking down at a laptop keyboard with a distressed or angry expression. The background is a warm, reddish-orange color.

Zo verlies je klanten.

**Hoe ga je daar mee om?**

**Geen prioriteit.**

# Tip 1

**Begin klein.**

A screenshot of the CodeSandbox web application. The interface includes a sidebar with navigation icons, a file tree on the left, and a main workspace on the right. A red box highlights a modal window titled "Adjust privacy settings." which shows the current setting is "Private".

Drafts / busy-wu-putzkt

Adjust privacy settings.

Private

Only you can see this Sandbox.

```
<div className="App">
  <h1>Hello CodeSandbox</h1>
  <h2>Start editing to see some magic happen!</h2>
</div>
```

Share Fork Create

Browser Tests Terminal

https://putzkt.

# Hello CodeSandbox

Start editing to see some magic happen!

A screenshot of the CodeSandbox web application. The interface includes a top navigation bar with tabs for 'Share', 'Fork', 'Create', and a user profile. Below the navigation is a sidebar with icons for files, dependencies, and other tools. The main workspace shows a file tree on the left with 'public', 'src' (containing 'App.js', 'index.js', 'styles.css'), and 'package.json'. A modal window titled 'Adjust privacy settings.' is open, showing a dropdown menu set to 'Private'. The modal also contains the text 'Only you can see this Sandbox.' and a preview of the application's output on the right. The preview shows a large 'Hello CodeSandbox' heading and a sub-heading 'Start editing to see some magic happen!'. The application code in 'App.js' is displayed in the center of the workspace.

```
<div className="App">
  <h1>Hello CodeSandbox</h1>
  <h2>Start editing to see some magic happen!</h2>
</div>
```

# Tip 2

**Gebruik de juiste <HTML />.**

A screenshot of a dark-themed code editor window titled "slide.html". The file path is shown as "Users > tristan > Desktop > slide.html". The code editor displays the following HTML content:

```
1  <button>Volgende slide</button>
2
3  <ul>
4  |  <li>Tip over klein beginnen</li>
5  |  <li>Tip over de juiste elementen gebruiken</li>
6  |  <li>Tip over geen spoilers sorry</li>
7  </ul>
8
9  <form>
10 |  <label for="asdf">Naam</label>
11 |  <input type="text" id="asdf" />
12 |  <button type="submit">Verzenden</button>
13 </form>
14
15
```

The editor has a sidebar on the left with various icons: file, search, file tree, refresh, and user settings. The status bar at the bottom shows "Ln 15, Col 1" and other file details.

Géén role="button"

# ARIA Authoring Practices Guide (APG)

How to build accessibility semantics into web patterns and widgets

**W3C**<sup>®</sup> Web Accessibility  
Initiative WAI

APG Home Patterns Practices Index About

## Patterns



### Read This First

No ARIA is better than Bad ARIA. Before using any ARIA, [read this to understand why.](#)



### Accordion (Sections With Show/Hide Functionality)

An accordion is a vertically stacked set of interactive headings that each contain a title, content



### Alert

An alert is an element that displays a brief, important message in a way that attracts the user's attention without interrupting the user's task.



### Alert and Message Dialogs

An alert dialog is a modal dialog that interrupts the user's workflow to communicate an important message and acquire a response.

# Chat GPT



A screenshot of a web browser window titled "HTML-tips van de AI." The address bar shows "chat.openai.com/chat". The main content area displays a conversation:

**T** Waarom ben je zo slecht in het schrijven van toegankelijke html?

 Als AI-taalmodel ben ik niet in staat om HTML te schrijven, omdat ik alleen in tekst kan communiceren. Bovendien ben ik geprogrammeerd om zo duidelijk en toegankelijk mogelijk te communiceren, dus ik doe mijn best om mijn antwoorden begrijpelijk en toegankelijk te maken voor iedereen.

# Tip 3

## Afbeeldingen

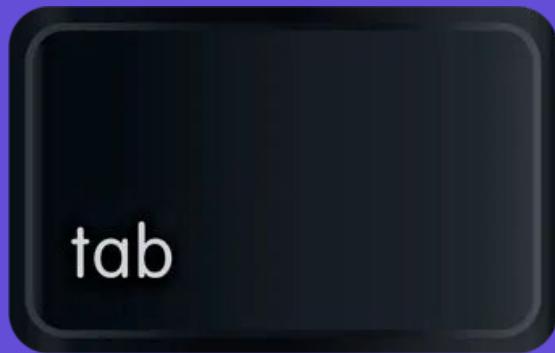
A screenshot of a dark-themed code editor window titled "slide.html". The file path is shown as "Users > tristan > Desktop > slide.html > ...". The code editor displays the following HTML:

```
1
2
3
4
5
6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
8 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
9 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
10
  <title>Decorative icon</title>
</svg>
```

```
<svg>
  <title>Circle</title>
  <desc>
    I'm a circle and that description
    is here to demonstrate how I can
    be described, but is it really
    necessary to describe a simple
    circle like me?
  </desc>
</svg>
```

# Laatste tip

**Gebruik je toetsenbord.**



**Meer tijd voor a11y?**

**Doe research!**

A, AA, AAA

**Terug naar de privacy picker.**

A screenshot of the CodeSandbox web application interface. The main window shows a dark-themed code editor with a modal dialog overlaid. The modal is titled "Adjust privacy settings." and contains a dropdown menu set to "Private". Below the dropdown, it says "Only you can see this Sandbox." The background of the editor shows a simple React component with an H1 and an H2.

The sidebar on the left is titled "CodeSandbox" and includes sections for "Files", "Dependencies", and "Add Dependency". The "Files" section lists "public", "src" (which contains "App.js", "index.js", and "styles.css"), and "package.json". The "Dependencies" section lists "loader-utils" (3.2.1), "react" (18.2.0), "react-dom" (18.2.0), and "react-scripts" (5.0.1). The "Add Dependency" input field has "Add Dependency" placeholder text.

The top navigation bar includes "Share", "Fork", "Create", and other standard browser controls. The URL in the address bar is <https://codesandbox.io/s/busy-wu-putzkt>. The right side of the interface features a preview area with the heading "Hello CodeSandbox" and the sub-instruction "Start editing to see some magic happen!".

```
<div className="App">
  <h1>Hello CodeSandbox</h1>
  <h2>Start editing to see some magic happen!</h2>
</div>
```

W3 Understanding Success Criteria X +

w3.org/WAI/WCAG22/Understanding/on-input.html

WCAG 2.2 Understanding Docs | *Informative explanations, not required to meet WCAG*

About WCAG Understanding Docs

W3C Web Accessibility Initiative WAI

All Understanding Docs ↑ Guideline: Predictable ← Previous SC: On Focus Next SC: Consistent Navigation →

Understanding SC 3.2.2:  
**On Input (Level A)**

**Success Criterion (SC)**

Changing the setting of any user interface component does not automatically cause a change of context unless the user has been advised of the behavior before using the component.

**Page Contents**

Intent  
Benefits  
Examples  
Techniques  
Key Terms

**Intent**

The intent of this Success Criterion is to ensure that entering data or selecting a form control has predictable effects. Changing the setting of any user interface component is changing some aspect in the control that will persist when the user is no longer interacting with it. So checking a checkbox, entering text into a text field, or changing the selected option in a list control changes its setting, but activating a link or a button does not. Changes in context can confuse users who do not easily perceive the change or are easily distracted by changes. Changes of context are appropriate only when it is clear that such a change will happen in response to the user's action.

**Note**

This Success Criterion covers changes in context due to changing the setting of a control. Clicking on links or tabs in a tab control is activating the control, not changing the setting of that control.

**Note**

W3 Understanding Success Criteria X +

w3.org/WAI/WCAG22/Understanding/on-input.html

WCAG 2.2 Understanding Docs | *Informative explanations, not required to meet WCAG*

About WCAG Understanding Docs

W3C Web Accessibility Initiative WAI

W3 How to Meet WCAG (Quickref) +

w3.org/WAI/WCAG21/quickref/#on-input

All Understanding Docs

Selected Filters: WCAG 2.1: all success criteria and all techniques.

Clear filters  Expand all sections  Share

Contents Filter Hide

2.5 Input Modalities

- 2.5.1 Pointer Gestures
- 2.5.2 Pointer Cancellation
- 2.5.3 Label in Name
- 2.5.4 Motion Actuation
- 2.5.5 Target Size
- 2.5.6 Concurrent Input Mechanisms

3. Understandable

- 3.1 Readable
  - 3.1.1 Language of Page
  - 3.1.2 Language of Parts
  - 3.1.3 Unusual Words
  - 3.1.4 Abbreviations
  - 3.1.5 Reading Level
  - 3.1.6 Pronunciation
- 3.2 Predictable
  - 3.2.1 On Focus
  - 3.2.2 On Input**
  - 3.2.3 Consistent Navigation
  - 3.2.4 Consistent Identification
  - 3.2.5 Change on Request
- 3.3 Input Assistance
  - 3.3.1 Error Identification
  - 3.3.2 Labels or Instructions
  - 3.3.3 Error Suggestion
  - 3.3.4 Error Prevention (Legal, Financial, Data)
  - 3.3.5 Help
  - 3.3.6 Error Prevention (All)
- 3.4 Robust
- 4.1 Compatible

Understandability

On Input

Success Criteria

Change has been made

Intended

The intention is to change the control or perception of the user

Note

This section tab contains notes

Note

## 3.2.2 On Input — Level A

Changing the setting of any user interface component does not automatically cause a change of context unless the user has been advised of the behavior before using the component.

Hide techniques and failures for 3.2.2  Sufficient  Advisory  Failures

### Sufficient Techniques

Note: Other techniques may also be sufficient if they meet the success criterion. See [Understanding Techniques](#).

- G80: Providing a submit button to initiate a change of context
  - H32: Providing submit buttons
  - H84: Using a button with a select element to perform an action
  - FLASH4: Providing submit buttons in Flash
  - PDF15: Providing submit buttons with the submit-form action in PDF forms
  - SL10: Implementing a Submit-Form Pattern in Silverlight
- G13: Describing what will happen before a change to a form control that causes a change of context to occur is made
- SCR19: Using an onchange event on a select element without causing a change of context

Note: A change of content is not always a change of context. This success criterion is automatically met if changes in content are not also changes of context.

### Advisory Techniques

- G201: Giving users advanced warning when opening a new window

### Failures

- F36: Failure of Success Criterion 3.2.2 due to automatically submitting a form and giving a value
- F37: Failure of Success Criterion 3.2.2 due to launching a new window without prior warning when the selection of a radio button, check box or select list is changed

## Bonus tip

Leer hoe een **screen reader** werkt.

W5 Understanding WCAG 2.1 | WAI X +

w3.org/WAI/WCAG21/Understanding/

WCAG 2.1 Understanding Docs

Informative explanations, not required to meet WCAG

About WCAG Understanding Docs

W3C Web Accessibility Initiative WAI

All Understanding Docs

# All WCAG 2.1 Understanding Docs

**Summary**

*Understanding documents provide detailed success criteria. They are informative, not part of the WCAG 2.1 standard.*

[Understanding Documents](#)

**Perceivable**

**Landmarks**

Understanding Docs navigation  
complementary  
navigation  
main  
complementary  
Page footer  
Site footer

**Page Contents**

Perceivable

Operable

Understandable

Robust

Other Understanding Documents

**1.1 Text Alternatives**

- [1.1.1 Non-text Content](#)

**1.2 Time-based Media**

- [1.2.1 Audio-only and Video-only \(Prerecorded\)](#)
- [1.2.2 Captions \(Prerecorded\)](#)
- [1.2.3 Audio Description or Media Alternative \(Prerecorded\)](#)
- [1.2.4 Captions \(Live\)](#)
- [1.2.5 Audio Description \(Prerecorded\)](#)
- [1.2.6 Sign Language \(Prerecorded\)](#)

W5 Understanding WCAG 2.1 | WAI X +

w3.org/WAI/WCAG21/Understanding/

## WCAG 2.1 Understanding Docs

All Understanding Docs

# All WCAG 2.1 Understanding Documents

### Summary

*Understanding documents provide detailed information about how to meet WCAG success criteria. They are informative, not prescriptive.*

### Perceivable

#### 1.1 Text Alternatives

- 1.1.1 Non-text Content

#### 1.2 Time-based Media

- 1.2.1 Audio-only and Video-only (Prerecorded)
- 1.2.2 Captions (Prerecorded)
- 1.2.3 Audio Description or Media Alternative (Prerecorded)
- 1.2.4 Captions (Live)
- 1.2.5 Audio Description (Prerecorded)

Informative explanations, not required to meet WCAG success criteria

## Headings

- 3: 3.3 Input Assistance
- 2: Robust
- 3: 4.1 Compatible
- 2: Other Understanding documents
- 1: All WCAG 2.1 Understanding Docs
- 2: Perceivable
- 3: 1.1 Text Alternatives
- 3: 1.2 Time-based Media
- 3: 1.3 Adaptable
- 3: 1.4 Distinguishable
- 2: Operable
- 3: 2.1 Keyboard Accessible
- 3: 2.2 Enough Time
- 3: 2.3 Seizures and Physical Reactions
- 3: 2.4 Navigable
- 3: 2.5 Input Modalities
- 2: Understandable
- 3: 3.1 Readable
- 3: 3.2 Predictable

WCAG® Web Accessibility Initiative WAI

### Page Contents

- Perceivable
- Operable
- Understandable
- Robust
- Other Understanding Documents

<https://www.w3.org/WAI/WCAG21/Understanding/> 1.2.1 Audio-only and Video-only (Prerecorded)

W3C Understanding WCAG 2.1 | WAI X +

w3.org/WAI/WCAG21/Understanding/

WCAG 2.1 Understanding Docs

Informative explanations, not required to meet WCAG

About WCAG Understanding Docs

W3C® Web Accessibility Initiative WAI

All Understanding Docs

# All WCAG 2.1 Understanding Docs

**Summary**

*Understanding documents provide detailed success criteria. They are informative, not part of the WCAG 2.1 standard.*

**Perceivable**

**Level 2 Headings**

- 2: Perceivable
- 2: Operable
- 2: Understandable
- 2: Robust
- 2: Other Understanding documents

**Page Contents**

- Perceivable
- Operable
- Understandable
- Robust
- Other Understanding Documents

**1.1 Text Alternatives**

- [1.1.1 Non-text Content](#)

**1.2 Time-based Media**

- [1.2.1 Audio-only and Video-only \(Prerecorded\)](#)
- [1.2.2 Captions \(Prerecorded\)](#)
- [1.2.3 Audio Description or Media Alternative \(Prerecorded\)](#)
- [1.2.4 Captions \(Live\)](#)
- [1.2.5 Audio Description \(Prerecorded\)](#)

<https://www.w3.org/WAI/WCAG21/Understanding/> [age \(Prerecorded\)](#)

W3C Understanding WCAG 2.1 | WAI X +

w3.org/WAI/WCAG21/Understanding/

WCAG 2.1 Understanding Docs

Informative explanations, not required to meet WCAG

About WCAG Understanding Docs

W3C® Web Accessibility Initiative WAI

All Understanding Docs

# All WCAG 2.1 Understanding Docs

**Summary**

*Understanding documents provide detailed explanations for Web Content Accessibility Guidelines (WCAG) guidelines and success criteria. They are informative, not part of the WCAG success criteria.*

**Level 4 Headings**

**Perceivable**

**1.1 Text Alternatives**

- 1.1.1 Non-text Content

**1.2 Time-based Media**

- 1.2.1 Audio-only and Video-only (Prerecorded)
- 1.2.2 Captions (Prerecorded)
- 1.2.3 Audio Description or Media Alternative (Prerecorded)
- 1.2.4 Captions (Live)
- 1.2.5 Audio Description (Prerecorded)
- 1.2.6 Sign Language (Prerecorded)

**Page Contents**

Perceivable

Operable

Understandable

Robust

Other Understanding Documents

jobs, the work, all of it. It's completely meaningless. I also believe this is a wonderful thing.

[READ THIS POST →](#)

### Turn 1 piece of dev content into 10+ — use the buffalo stick

Learn how the best developer experience engineers multi-leverage ideas to turn one piece of content into 10+. This is THE secret to effective devrel.

[READ THIS POST →](#)

Burned out to enjoy your success)? You need to apply constant, gentle pressure.

[READ THIS POST →](#)

deliberate practice and continuous feedback. Small improvements add up fast.

[READ THIS POST →](#)

was wrong. Ability doesn't matter if you don't do the work.

[READ THIS POST →](#)

### Please stop multitasking. I'm begging you. Please.

The greatest trick we ever pulled on ourselves as knowledge workers was convincing ourselves we could juggle multiple projects with no consequences.

[READ THIS POST →](#)

### Be weird & fun (as a business strategy)

Every once in a while you get lucky enough to work on a project that taps into everything you love. I led one recently — and it goes live TODAY.

[READ THIS POST →](#)

### Output vs. Outcome

When you define work, do you focus on the output or the outcome? Learn what the difference is — and why it matters.

[READ THIS POST →](#)

### Checking Slack and email during vacation is bad for you AND your team. Please stop doing it.

Taking real, fully disconnected time off is good for your health and for your team. But you have to ACTUALLY disconnect to get the benefits.

[READ THIS POST →](#)

### The defaults matter

No matter what you're building — habit, tool, or company culture — choosing the right defaults is critical. Learn how to use inertia to your advantage.

[READ THIS POST →](#)

### Intention vs. Drift

There are two ways to create. Either we intentionally make progress toward goals, or we drift & see what happens. For best results, I think we need both.

[READ THIS POST →](#)

### Don't Be Nice. Be Kind.

"If you can't say something nice, don't say anything at all" is terrible advice that often backfires. Let's find another way of looking at caring for people.

[READ THIS POST →](#)

### How to Draw a Map to a More Fulfilling Career

If we want a happy, fulfilling

### More pie

They say "no good deed goes unpunished". When we finish

### Convince your team to learn fast and try not to make a mess

### Yeet. Delete. Repeat.

We need to ship fast, learn fast — and do it all without



**Veel frontend.**

Design.

# Backend.

**CMS inrichten voor screen reader  
specifieke teksten.**

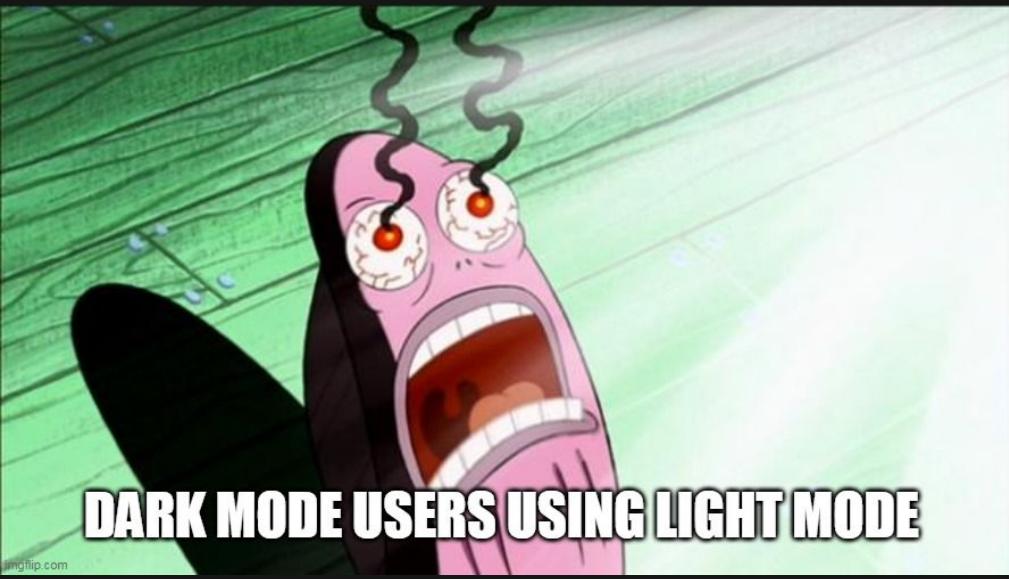
Multi language.

**Help content managers.**

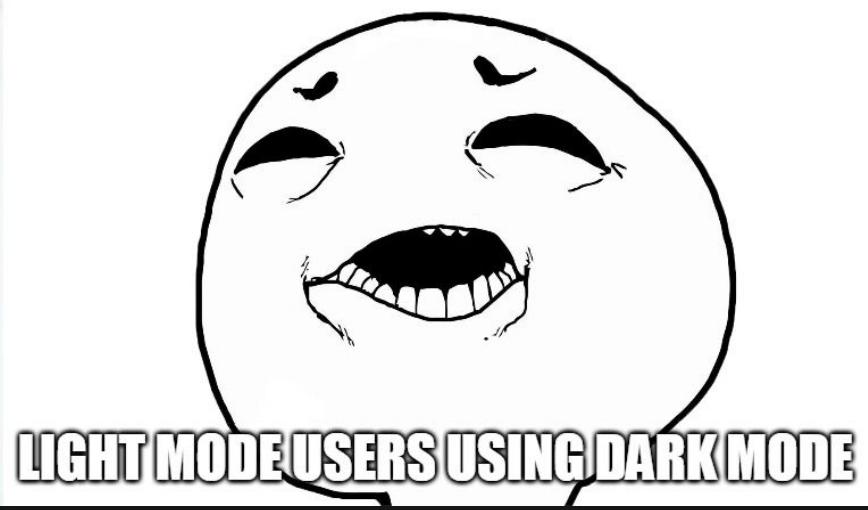
**Overtuig project management om er iets  
aan te doen.**

Voor iedere discipline geldt:

**Niet alles tegelijk.**



**DARK MODE USERS USING LIGHT MODE**



**LIGHT MODE USERS USING DARK MODE**

**Accessibility  
is voor iedereen.**

**Kleine verbeteringen,  
groot effect.**

**Begin.**