

The power of progressive enhancement

Hi there, I'm Andy 🖐️

I'm a designer and front-end
developer from the U.K.



I'm an accessibility and progressive
enhancement super-fan 🚀.

The main reason I care a lot about
these things...

People

People are who we build for

Why, though?

Let's start with something
very important

Fast, reliable internet is a
LUXURY, not the standard

Average internet speeds: *Developed Countries*

 Italy: **9.2mbs**

 France: **10.8mbs**

 Australia: **11.1mbs**

Source: https://en.wikipedia.org/wiki/List_of_countries_by_Internet_connection_speeds

<https://hankchizljaw.io>

@hankchizljaw

For context: 3G is about 7mb/s

Surprised at how slow they are?

Average internet speeds: *Developing Countries*

 Paraguay: 1.4mbs

 Nigeria: 3.9mbs

 India: 6.5mbs

Source: https://en.wikipedia.org/wiki/List_of_countries_by_Internet_connection_speeds

<https://hankchizljaw.io>

@hankchizljaw

The average page weight
on desktop is **1526kb**

Source: <https://httparchive.org/reports/page-weight>

<https://hankchizljaw.io>

@hankchizljaw

This isn't a talk about
performance, though.

Well, not directly.

You can improve performance with
progressive enhancement

Let's have a chat about devices

This might come as a surprise*

**(it shouldn't)*



Photo by Stephen Frank on Unsplash

Android has ~ 70% market share

Source: <http://gs.statcounter.com/os-market-share/mobile/europe>

<https://hankchizljaw.io>

@hankchizljaw

A lot of people browse the web
without JavaScript

Not necessarily because
“they don’t like it”

Some reasons

There was a runtime error

It's disabled as the
ultimate Ad Blocker

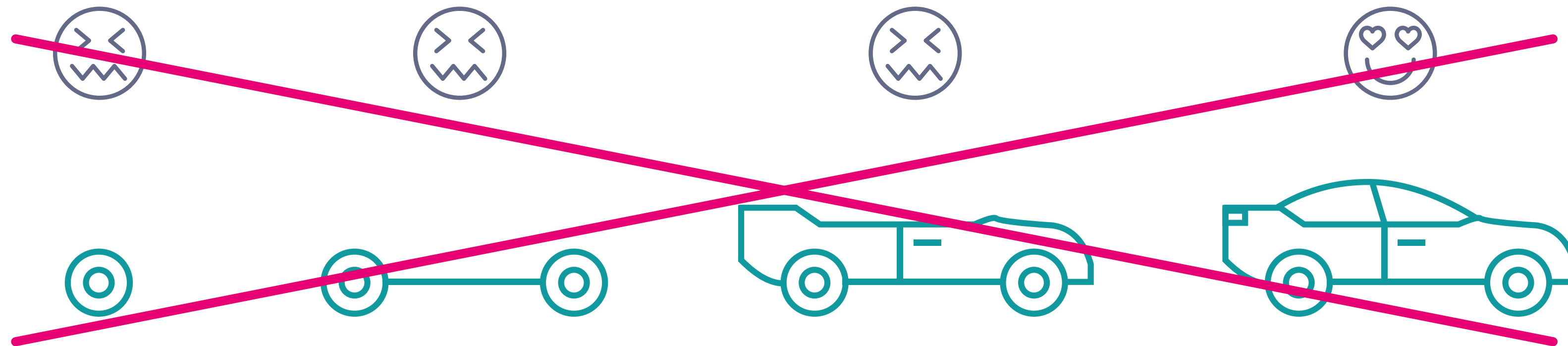
They run Android on 2G
with data-saver enabled

How does progressive enhancement work?

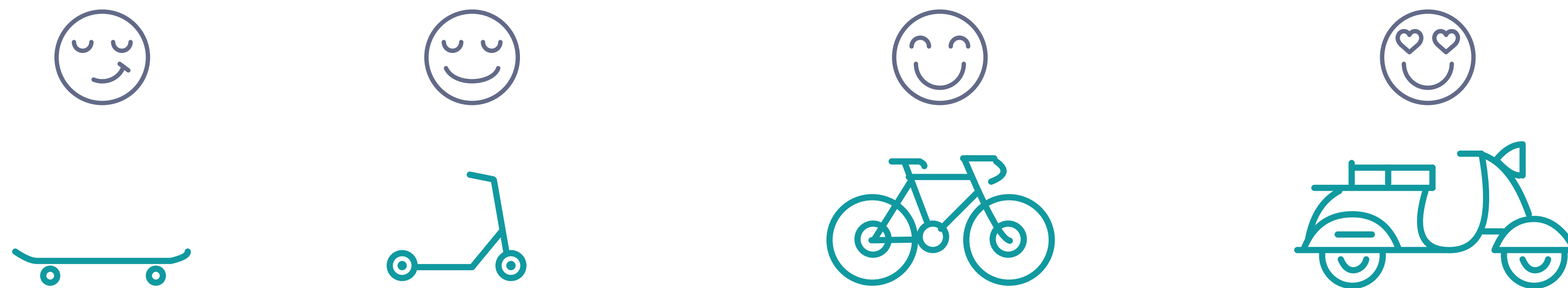
Minimum Viable Experience 🎉

What the heck is that?

Not like this

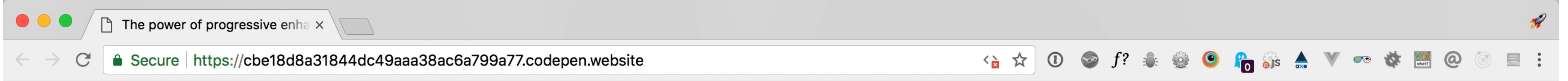


Like this!



Provide the most possible value
to a user with the least possible
technical capability

A quick example



Progressively enhanced toggle panel

By default, this is a heading and some paragraphs, but where web components are supported, a toggle panel is available.



A heading which toggle the collapsable panel

Cras mattis consectetur purus sit amet fermentum. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Duis mollis, est non commodo luctus, nisi erat porttitor ligula, eget lacinia odio sem nec elit. Nullam quis risus eget urna mollis ornare vel eu leo. Aenean eu leo quam. Pellentesque ornare sem lacinia quam venenatis vestibulum. Curabitur blandit tempus porttitor. Nullam id dolor id nibh ultricies vehicula ut id elit.

Etiam porta sem malesuada magna mollis euismod. Donec sed odio dui. Curabitur blandit tempus porttitor. Sed posuere consectetur est at lobortis. Aenean lacinia bibendum nulla sed consectetur. Nullam quis risus eget urna mollis ornare vel eu leo. Integer posuere erat a ante venenatis dapibus posuere velit aliquet.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam id dolor id nibh ultricies vehicula ut id elit. Praesent commodo cursus magna, vel scelerisque nisl consectetur et. Maecenas faucibus mollis interdum. Aenean lacinia bibendum nulla sed consectetur.

The priority is the content, which is
delivered immediately

The fancy interaction doesn't have to
work for the user to get any value

The user will never know 🎉

Minimum Viable Experience 🎉

<https://hankchizljaw.io>

@hankchizljaw

A common response as to why
progressive enhancement
can't be considered

“The users need the exact, same experience in every browser”

- *Someone in marketing, probably*

Another response

“I can’t possibly do that with my
complex JavaScript application”

- *Some tech bro, probably*


This one is probably true.

Progressive enhancement is a
mindset change

Progressive enhancement isn't necessarily *more work* and it certainly isn't a *non-JavaScript fallback*, it's a change in how we think about our projects. A complete mindset change is required here and it starts by remembering that you don't build websites for *yourself*, you build them for *others*.

Embrace the web platform 🧊

Simple setup = focus on what's
important

 Frameworks 



Frameworks



Progressive enhancement + Frameworks =



What *free* tools are these?

Web Components and CSS 🎉

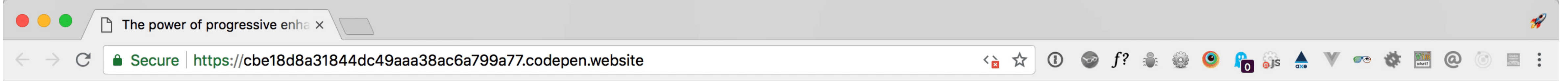
“But not all browsers support
Web Components”

- *That person on Twitter*

No fallbacks, thanks



Let's look back at our example



Progressively enhanced toggle panel

By default, this is a heading and some paragraphs, but where web components are supported, a toggle panel is available.



A heading which toggle the collapsable panel

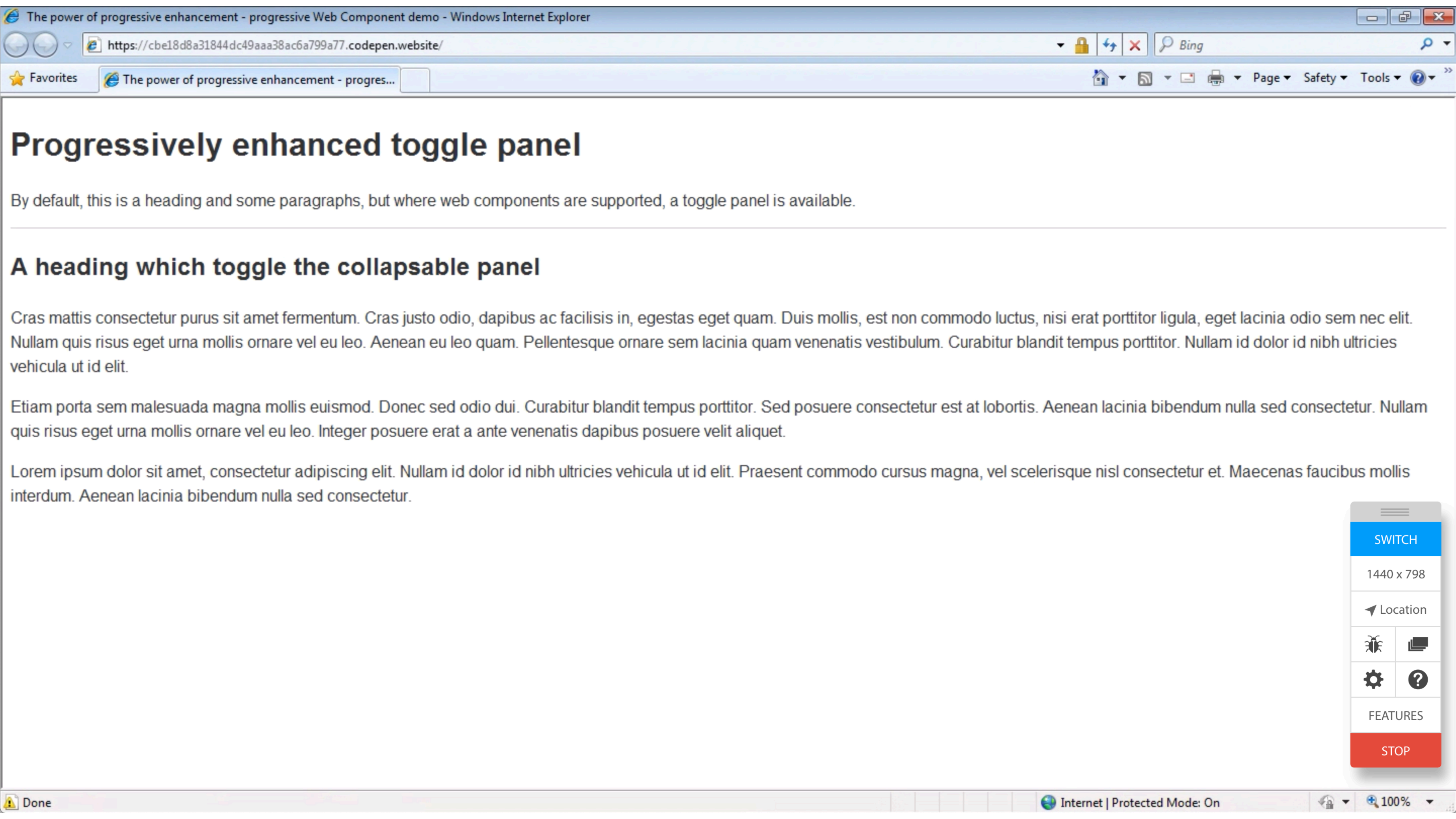
Cras mattis consectetur purus sit amet fermentum. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Duis mollis, est non commodo luctus, nisi erat porttitor ligula, eget lacinia odio sem nec elit. Nullam quis risus eget urna mollis ornare vel eu leo. Aenean eu leo quam. Pellentesque ornare sem lacinia quam venenatis vestibulum. Curabitur blandit tempus porttitor. Nullam id dolor id nibh ultricies vehicula ut id elit.

Etiam porta sem malesuada magna mollis euismod. Donec sed odio dui. Curabitur blandit tempus porttitor. Sed posuere consectetur est at lobortis. Aenean lacinia bibendum nulla sed consectetur. Nullam quis risus eget urna mollis ornare vel eu leo. Integer posuere erat a ante venenatis dapibus posuere velit aliquet.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam id dolor id nibh ultricies vehicula ut id elit. Praesent commodo cursus magna, vel scelerisque nisl consectetur et. Maecenas faucibus mollis interdum. Aenean lacinia bibendum nulla sed consectetur.

The Web Component enables our
minimum viable experience

It works in IE8



Progressively enhanced toggle panel

By default, this is a heading and some paragraphs, but where web components are supported, a toggle panel is available.

A heading which toggle the collapsable panel

Cras mattis consectetur purus sit amet fermentum. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Duis mollis, est non commodo luctus, nisi erat porttitor ligula, eget lacinia odio sem nec elit. Nullam quis risus eget urna mollis ornare vel eu leo. Aenean eu leo quam. Pellentesque ornare sem lacinia quam venenatis vestibulum. Curabitur blandit tempus porttitor. Nullam id dolor id nibh ultricies vehicula ut id elit.

Etiam porta sem malesuada magna mollis euismod. Donec sed odio dui. Curabitur blandit tempus porttitor. Sed posuere consectetur est at lobortis. Aenean lacinia bibendum nulla sed consectetur. Nullam quis risus eget urna mollis ornare vel eu leo. Integer posuere erat a ante venenatis dapibus posuere velit aliquet.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam id dolor id nibh ultricies vehicula ut id elit. Praesent commodo cursus magna, vel scelerisque nisl consectetur et. Maecenas faucibus mollis interdum. Aenean lacinia bibendum nulla sed consectetur.

SWITCH

1440 x 798

Location

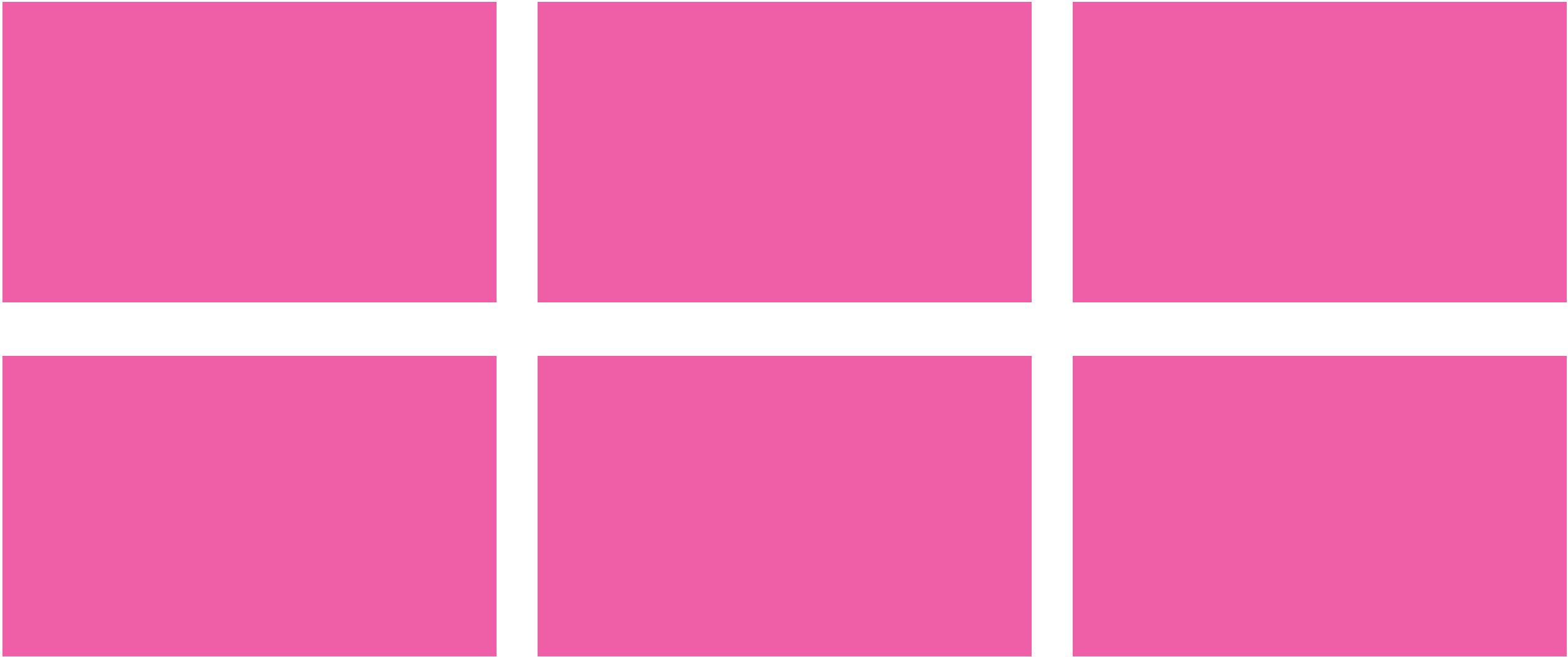
FEATURES

STOP

We're producing a resilient
front-end by default 🎉

How about CSS?

Planning is your pal



What's the minimum
viable experience?



Easy to consume.
Easy to navigate.

Minimum Viable Experience 🎉

Job done for the 16% of
global users that don't have
CSS Grid support.

Source: <https://caniuse.com/#feat=css-grid>

<https://hankchizljaw.io>

@hankchizljaw

What about the other 84%?

```
display: grid;  
grid-template-columns: repeat(auto-fill, minmax(16.5rem, 1fr));  
grid-gap: 1rem;
```

That's the full-blown experience


```
.grid {  
  display: grid;  
  grid-template-columns: repeat(auto-fill, minmax(16.5rem, 1fr));  
  grid-gap: 1rem;  
}
```

```
.grid > * {  
  max-width: 400px;  
}
```

```
.grid > * + * {  
  margin-top: 20px;  
}
```

Let's put heavy duty CSS
frameworks away

Progressive enhancement means
tiny amounts of CSS

We should embrace that. *All of us.*

What else can we do?

Make progressive enhancement a
part of every process in your team

- Planning
- UX
- Design
- Development
- Testing

“What is the minimum
viable experience and
how can we enhance it?”

Always remember, the priority it's
the **people** that use our websites.

- Not ourselves.
- Not our Twitter peers.
- Not Dribbble.

Developer convenience is
important, only if it frees us up
to focus on our users.

Focus on tools, frameworks &
hotness only: we fail to do our jobs.

Tools are good, but happy
users are better

A user doesn't care how
your thing was built

Recap

People are our main priority

Ditch the fallbacks, hacks
and heavy frameworks

Embrace a minimum
viable experience

Create a resilient,
user-focused experience

Sleep easy knowing your website or
app *just* works.

Thank you