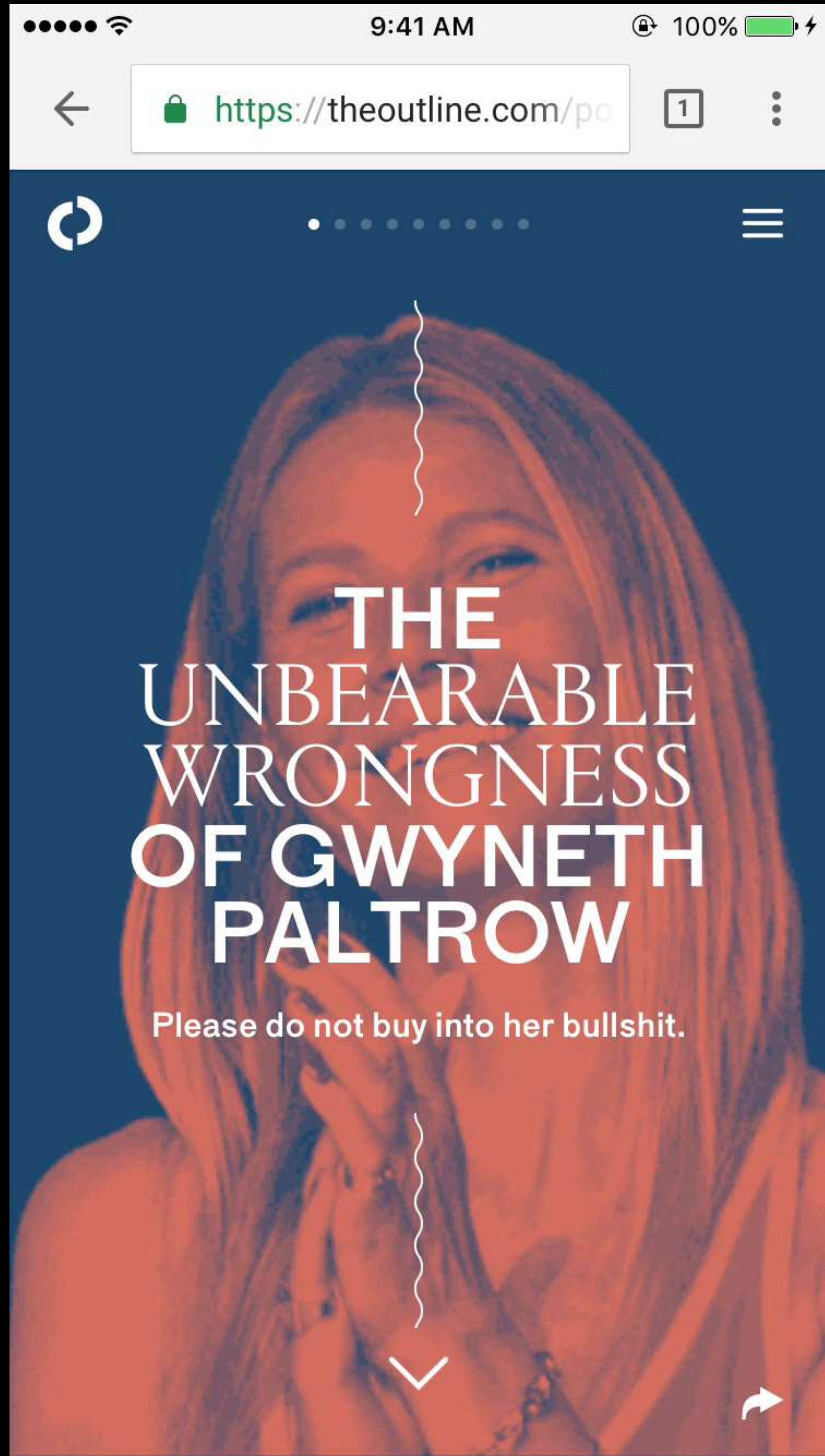


Visual storytelling at The Outline

Ivar Vong, CTO
@ivarvong

What is The Outline?

- Founded by Josh Topolsky, formerly The Verge, Bloomberg, etc
- Coverage areas: Power, Culture, and The Future
- Full-stack technology platform from scratch
- {mobile, open_web, visual} stories
- Launched in December 2016
- Build started 3 months before launch



Design System

- Design primitives shared between editorial and ad products
 - “Card Type” system
- Code shared between editorial and ad products
- Ad product: “No foreign objects”

How?

- Elixir/Phoenix
- ES6/React/Redux
- Postgres, Redis
- CI/CD on PaaS (Heroku)
- Small, cross-functional team, agile-ish
- Monorepo

...Elixir? ...Phoenix?

- Elixir
 - Functional, immutable, concurrent, pattern matching, macros
- BEAM is fast: 99th percentile, at load, <90ms HTTP response
- Phoenix: similar to Rails (* except no magic mutability)
- Great websocket support
- Sub-millisecond template rendering
- Attractive to senior dev talent

BEAM

- Preemptive scheduler
- Erlang/BEAM open source since 1998
- Much of Elixir is just Erlang

Edge

- No public CDN caching of HTML
 - Push as much logic as possible into Elixir app
 - Simplifies (otherwise complex) feature flags
 - (Static assets are digested and have long CDN cache lives)
- Server-render as much HTML as possible (HTML in JSON for prefetching)

HTTP/2

- HSTS preload
 - Privacy, integrity, performance
- Heroku (origin) is not HTTP/2, CloudFlare is
 - Origin <> CDN is HTTPS and HTTP/1.1
- HTTP/2 push CSS & JS via `Link` HTTP header + CloudFlare
- Pushing web fonts decreased initial page load performance

Front-end

- Touch navigation on mobile
- Built lightweight front-end JS app (ES6, Redux, but no React)
 - Had experience w/ Redux, and wanted known pattern to manage FE state
 - View lifecycle events (similar to native mobile), analytics, data loading
- React + server-rendered HTML = ???
- ...webpack....

CMS

- Storytelling tool: React/Redux app
- HTTP JSON API to Phoenix app
- Ecto custom types + internal DSL via macro
 - Ecto is data validation and persistence library, similar to ActiveRecord
 - JSONB in PG gives us NoSQL-ish but w/ Ecto validation system

CMS

- Declare schemas per content type, generate editing surface
- Combinatorial complexity yields 10,000+ variations
- Built by designers and developers, used by reporters and editors
 - “Color schemes, not color pickers”
 - Hard to make things that don’t look great
- Live preview increases experimentation by shortening feedback loop
- Live preview on real devices, over websocket

Use Vendors

- Cloudflare (supports websocket)
- Heroku (pipelines, preboot, 14GB VMs, immutable infra)
 - Librato, Papertrail
- Appsignal (error/perf reporting, has good Elixir integration)
- CircleCI
- Amplitude
- Redshift

Problems We Didn't Solve

- Template caching
- Message queue
- Deployment
- Universal Javascript
- Event loop
- Microservices
- Versioning APIs

Next

- Distributed Erlang?
 - Global process registry would be helpful
- Kubernetes? GKE?
- New story forms

Thanks!

- <https://theoutline.com>
- @ivarvong
- ivar@theoutline.com