Charles Gallant

UX / Product Design Director

2022 Portfolio

If you’re receiving this, it means I’d love to talk with you about my work,
and the journey thus far. Please feel free to reach out at charlesgallant@gmail.com

Thank you!

As a designer, I try to build amazing experiences based on logic, aesthetics, and most of all empathy.

As a coder, I study new frameworks and platforms to find opportunities for building smarter and better things.

As a human, I look for great people that I can learn from, who share in my goal of spreading positivity.

I'm a coder who became a designer. I specialize in solving complex problems.
I believe in a thoughtful relationship between design and technology.
Success to me is being a good person, and creating best in class product experiences.

The journey thus far...

2005 — 2013 — 2019 — 2022

Workplaces

NYC

Marketing & Digital Agencies
Working as an Engineer / Creative Technologist

SF

Startups & Product Companies
Working as an UX / Product Designer

CO

…A little of everything!
Working as Experience Director, Startup founder, and consultant.

Creative / Design Practices

Fine Art
Design
Animation

Marketing Strategy
UX Design

Product Strategy
Product Design

UX Direction
UX Consulting

Technology Practices

Front-end / Back-end
Web Engineering

Objective-C &
Mobile Frameworks

Unity & C#
Electronics

Dev Ops
Procedural Audio

More info, for the curious…

2005 — 2013 — 2019 — 2022

Workplaces

NYC

Marketing & Digital Agencies
Working as an Engineer / Creative Technologist

SF

Startups & Product Companies
Working as an UX / Product Designer

CO

…A little of everything!
Working as Experience Director, Startup founder, and consultant.

RENEGADE

Front-end + Back-end Engineer & Flash Animator
Renegade Marketing

Wieden+Kennedy

Creative Technologist
Wieden Kennedy, NYC

anki

Lead UX Designer
Anki

Experience Director
CP+B

Crispin Porter Bogusky

Lead UX Designer
Ozobot

POKE

Front-end Engineer, Flash Interaction Designer, UX Designer
Poke New York

West

UX Designer, Creative Technologist
West SF
(aka West Ventures)

Product Design Consultant
Contract

CartHook

Founder, CTO
CartHook.com

Company: Anki

Role: Lead UX Designer

Anki was a robotics & AI company making award winning smart toys. They're the best place to start, because my time there represented the most exciting and complicated challenges of my career to date.

During my time there, my responsibility was to lead UX Design across their 3 products: Drive, Overdrive, and Cozmo. Growing from Anki's sole UX designer to eventually leading the UX Design Team was an amazing journey. Along the way I dove deep into physical hardware, mobile UI, BLE connectivity, multiplayer game design, and the sleepless pace of SF startup life.

Drive

OverDrive

Cozmo

At a robotics company that creates a mixed hardware / software experience, hundreds of questions were always top of mind. In this environment, the job of UX Design was about understanding all dimensions of every question, parsing and weighing them against time & resources, and creating the simplest and best experience for the user. In many ways, UX Design was really synonymous with Product Design & Strategy.

Do users need to charge the cars first, or can they be used out of the box? Is our first onboarding step “Charge your cars!”?

In multiplayer scenarios, does one player need to explicitly define themselves as the Host?

How fast is a firmware update over BLE? What does the user see if the connection drops mid update?

Can Android and iOS users play together? Can the app know that they aren’t on the same wifi access point, and message accordingly?

If the sensors on the cars are having trouble reading the track codes, can we detect this?

How accurate is a car’s position awareness while it is motionless?

When is it appropriate to teach players how to switch weapons? How long does their first launch take, and when might they be overwhelmed?

How much do we hand-hold through a tutorial, and when do we let users tap around and explore on their own?

Does a car’s virtual upgrades “live” with the car, or on the user’s profile? What if their car breaks?

	Driver only Progress	Car+Driver Progress	Car only Progress
 **Driver**	**Driver abilities progress over time;** Points / XP is gained that affects all driving across all cars.  For example... **Top speed: 325** **Acceleration: 50** **Shield: 100** **Guns: 50**	**Driver abilities progress over time;** Points / XP is gained that affects all driving across all cars.  For example... **Shield: 100** **Guns: 50**	**Drivers don't build any special skills / XP of their own.** 
 **Car**	**Cars accumulate no "XP" or skills of their own.** Car's abilities are defined by the current driver. 	Cars build up XP or skills over time, which upgrade the car's abilities. **The overall driving capacity is defined by the *combined* driver and car "progress."**  For example... **Top speed: 325** **Acceleration: 50** *Optionally, "items" are collected by drivers and applied to cars, as non-permanent upgrades (like a special gun or shield).*	**The cars themselves build skills and XP over time.**  For example... **Top speed: 325** **Acceleration: 50** **Shield: 100** **Guns: 50**
	PROS: Cars are simpler to share and trade, less of an issue if they break. Progress can be applied to other parts of the Anki ecosystem. **CONS:** Cars might feel less special / owned / unique.	**PROS:** Creates attachment to cars, gamers are incentivized to build skills not just for their profiles but also all the cars. **CONS:** Slightly more complex to understand. The difference between gamer and car skills may not be evident.	**PROS:** Increased attachment to individual cars, increased incentive to collect cars. **CONS:** If a car malfunctions, progress is lost. Doesn't apply to other Anki products.

Unlike the years of UX work before it, design docs weren't always wireframes at Anki. In the early days, documents like this one were critical for helping the team visualize complex ideas, and make decisions. Here, we needed to discuss the strategy for player-based and vehicle-based progression systems, that would define the game's behavior loops.

Garage Car Detail View

Sub-pages:
- Add Upgrade View
- Equip Items View
- Multiple Vehicles View (potentially an expanding panel)

Level indicator identifies the progress made with this vehicle thus far, and what you could make in the future.

Images of latest item or upgrade is visible above respective buttons.

Buttons are badged (highlighted green, in this wireframe) when the user has something to do. For example, if new upgrades have been collected and can be equipped, these views would be highlighted.

Right-side vertically scrolling nav. First index (wrenches) is 'Garage Gome'

Currently selected vehicle is visibly different in the list.

This garage detail view is responsible for displaying two types of information:

Virtual Vehicle Information (top area): Information that applies to your virtual spec for this vehicle (un-related to wether or not you own that vehicle, or if one is present).

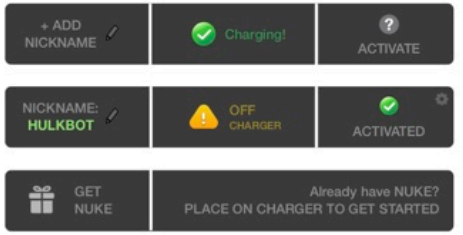
Physical Vehicle Information (bottom section): Information that applies only to the physical cars (of this type) that the app is aware of. This area would change dynamically if there are no cars available, or if we have never seen this particular car.

If there is more than one NUKE vehicle present, a special button would appear allowing you to see information about multiple nearby vehicles (dimmed here to not steal attention).

This may be a separate view with a simple list, each one showing the content displayed at left (name, charging status, active status).

If not (most use cases), this button would be hidden, and this area empty.

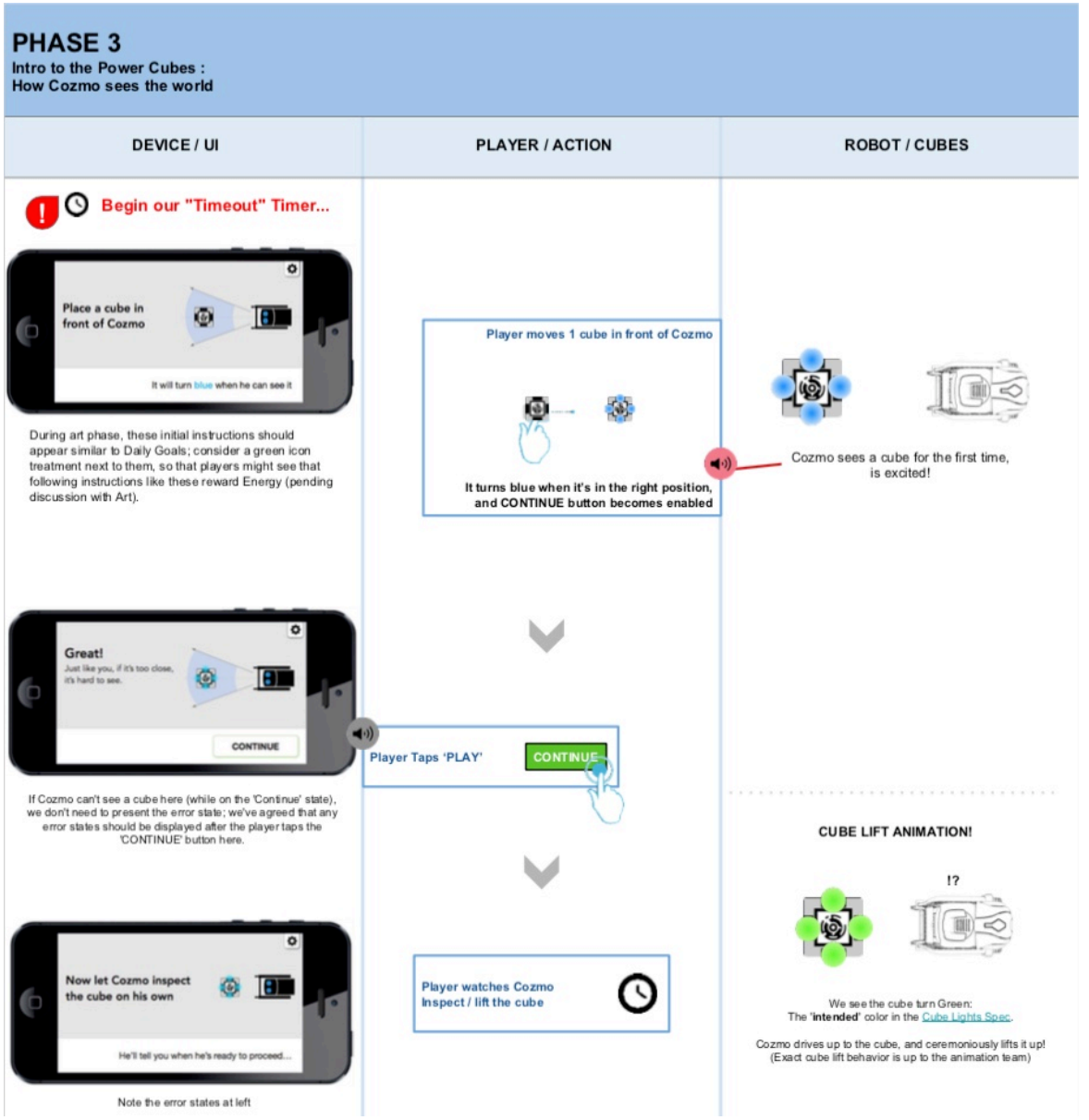
Some examples below. See following slide for details / use cases.



Players can add/edit their vehicle's nickname.

Players can modify ownership information; "de-activating" or "de-associating" this vehicle from their profile (although there is no reason to).

Always present were more traditional design documents, such as annotated wireframes and written functional specifications. The above wireframe was one of hundreds of views for our second gen car game, Overdrive. Here, the user sees their virtual representation of a physical car, revealing its in-game stats / weapons.



(These documents became Scrolls of Truth that grew very, very long!)

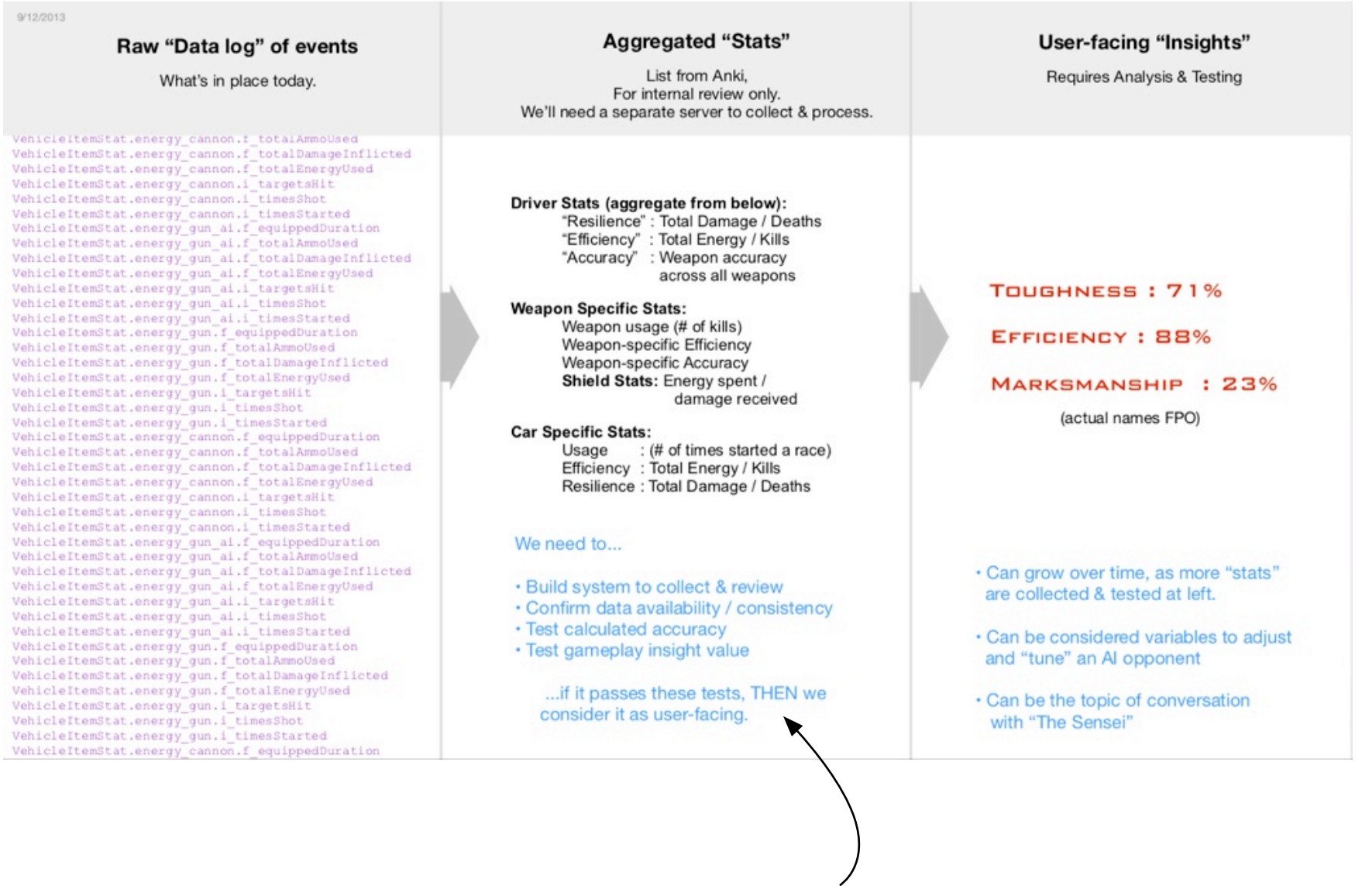
We quickly learned that traditional wireframes and user flows weren't enough, and we needed something that captured all 3 contexts:

App behavior, Human behavior, Robot behavior.

This excerpt from the **Cozmo Onboarding Design** shows a new type of design doc that became a requirement for all future Cozmo user flows.

At **left** are the in-app wireframes, the **middle** are the user's actions, and the **right** are the physical bot and cubes. All of this is read top-to-bottom, over time.

Hundreds of hours of iteration went into managing diagrams like these in order to capture all possible permutations of app / human / bot state.



I was entrusted by Anki to lead the UX Team because I could bridge the gap between the Engineering and the Design, and identify opportunities for UX value. Above was an example of this in practice: We wanted to turn raw analytics logs into user-facing stats, but only if we could first prove that they provided real in-game value.

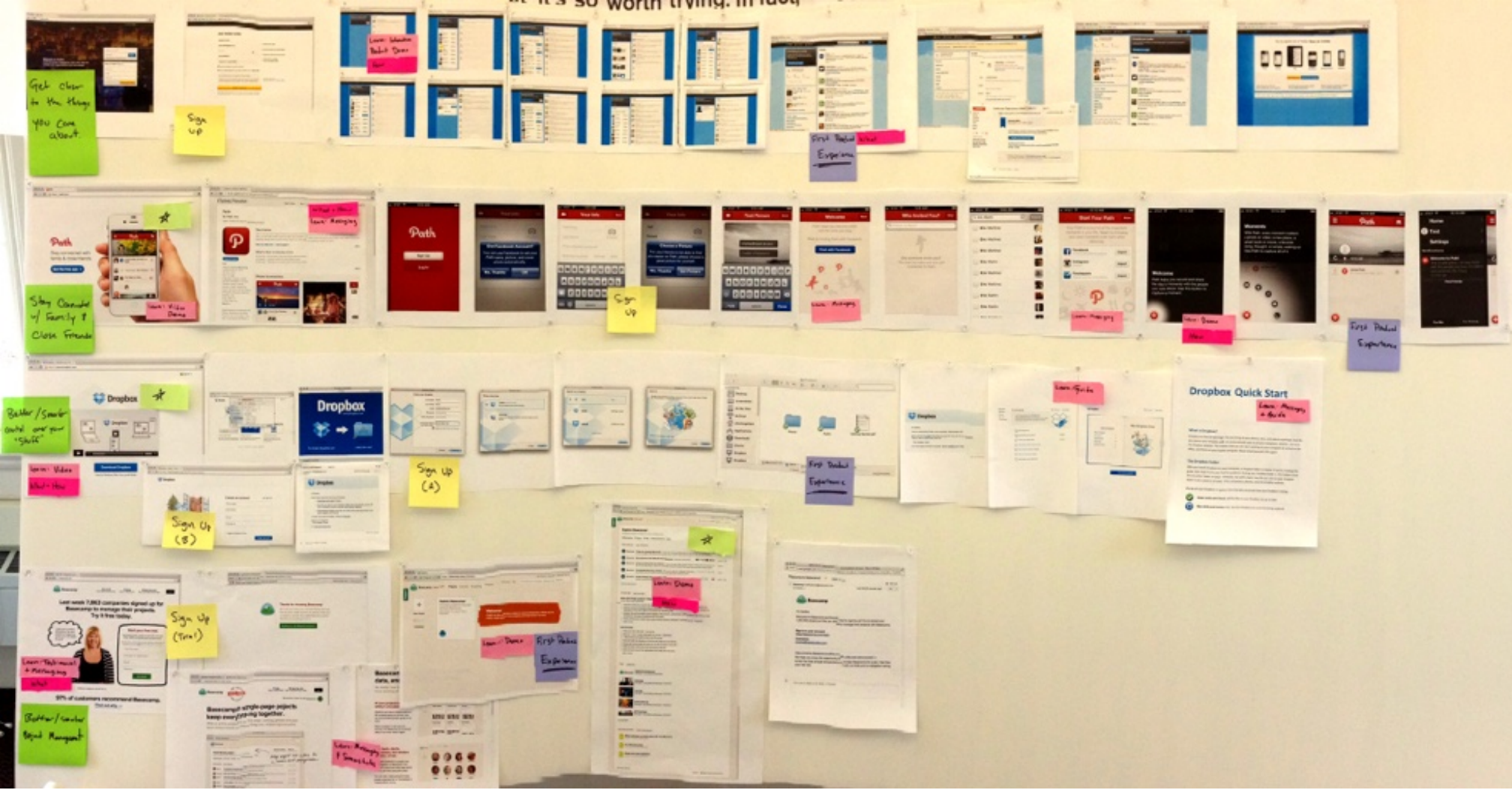
west

Company: West

Role: UX Designer / Creative Technologist

West was a multi-disciplinary creative collective that helps the world's most disruptive companies grow. Here I worked with some of the biggest brands in San Francisco to consult on onboarding flows, user acquisition techniques, marketing strategies, and product designs.

By the way, we’re jumping around in time here…

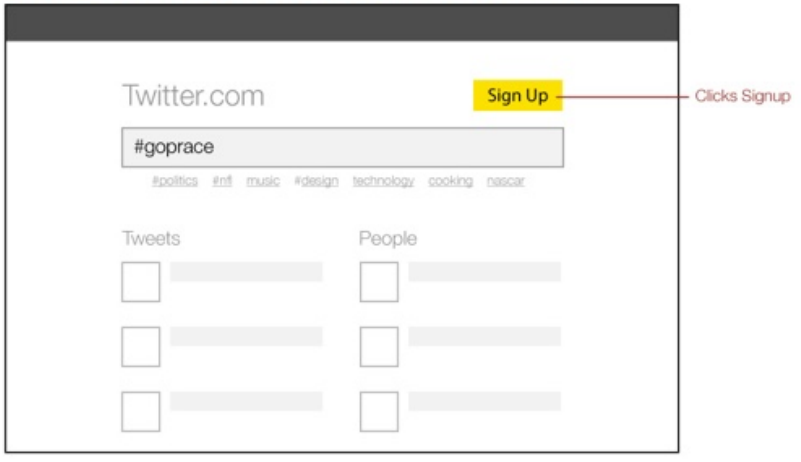


As UX Lead at West, one of my tasks was to become “The Onboarding Expert.”
What is the first thing that users see? What convinces them to sign up? What key things do they need to learn? And so on…

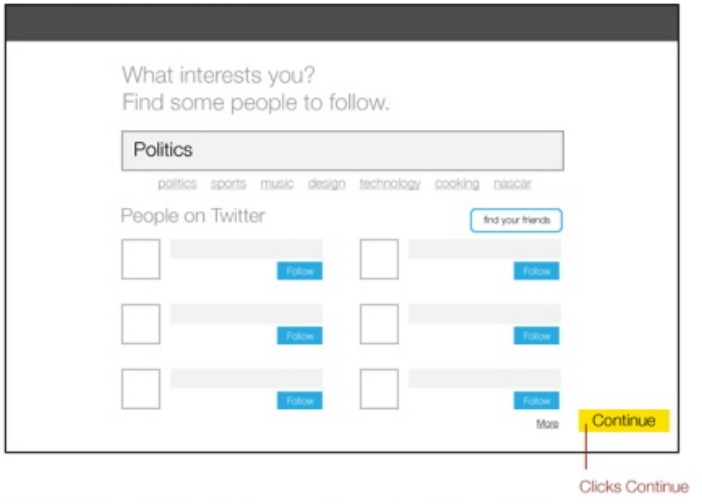
Many boards like this one were created — not only for West clients (Twitter, Path, Dropbox), but for the competitive spaces they occupied as well. From here, we tried to reverse engineer their product/messaging priorities, and created a ‘best practices’ playbook for onboarding tactics that we could pass along to our portfolio of clients.

Proposed Twitter Signup Flow

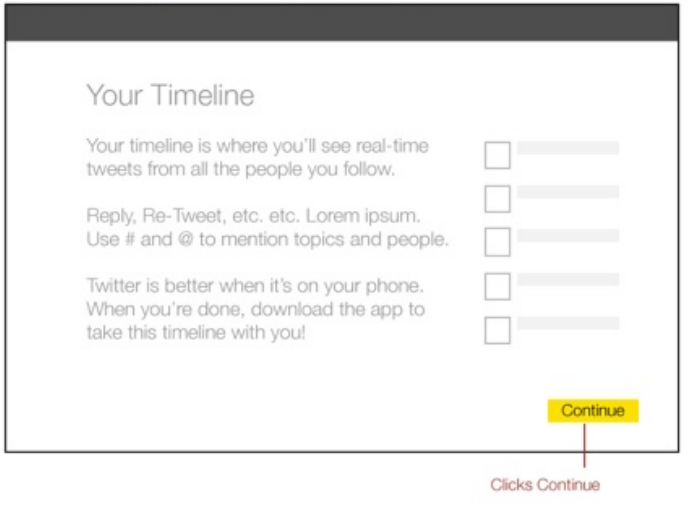
1. Twitter.com homepage: Invite Sign Up by demonstrating value.
Homepage should have more than empty signup fields and an image.



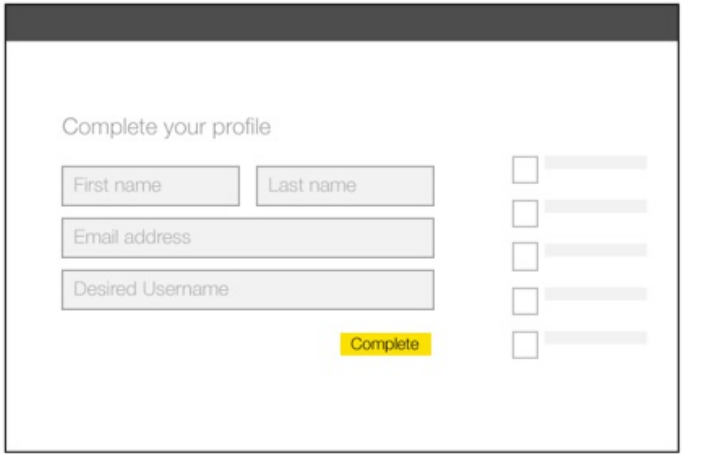
2. After clicking signup, connect them with their interests BEFORE forcing them to commit and enter all their info.



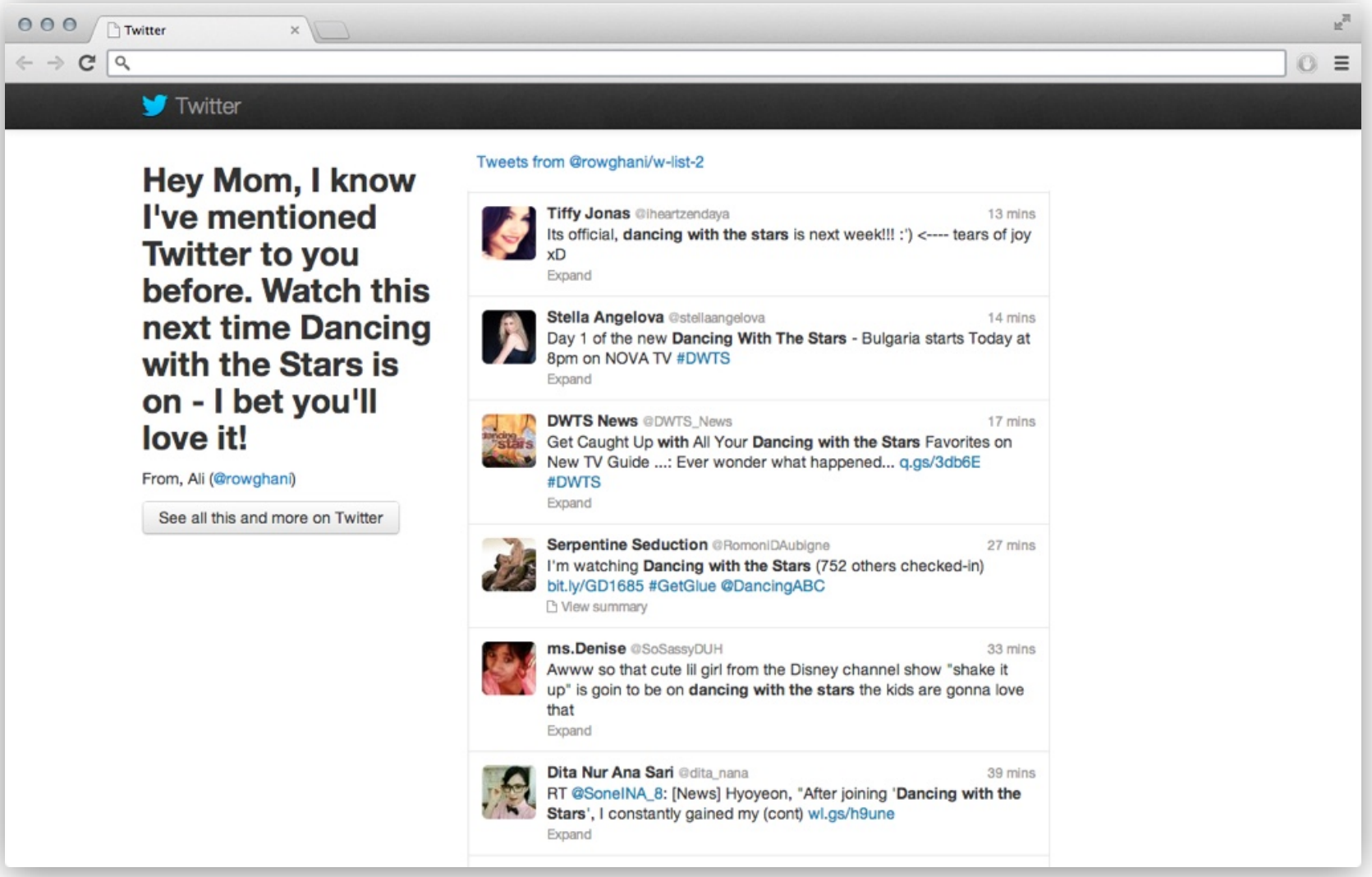
3. Show them exactly what Twitter offers (a timeline), and explain it. Use info gathered from the previous step to make this valuable & customized to the user. Language like "hear it first from [person followed]".



4. Ask for their info after they've seen the value of Twitter, not before. Keep the timeline visible here as a reminder.

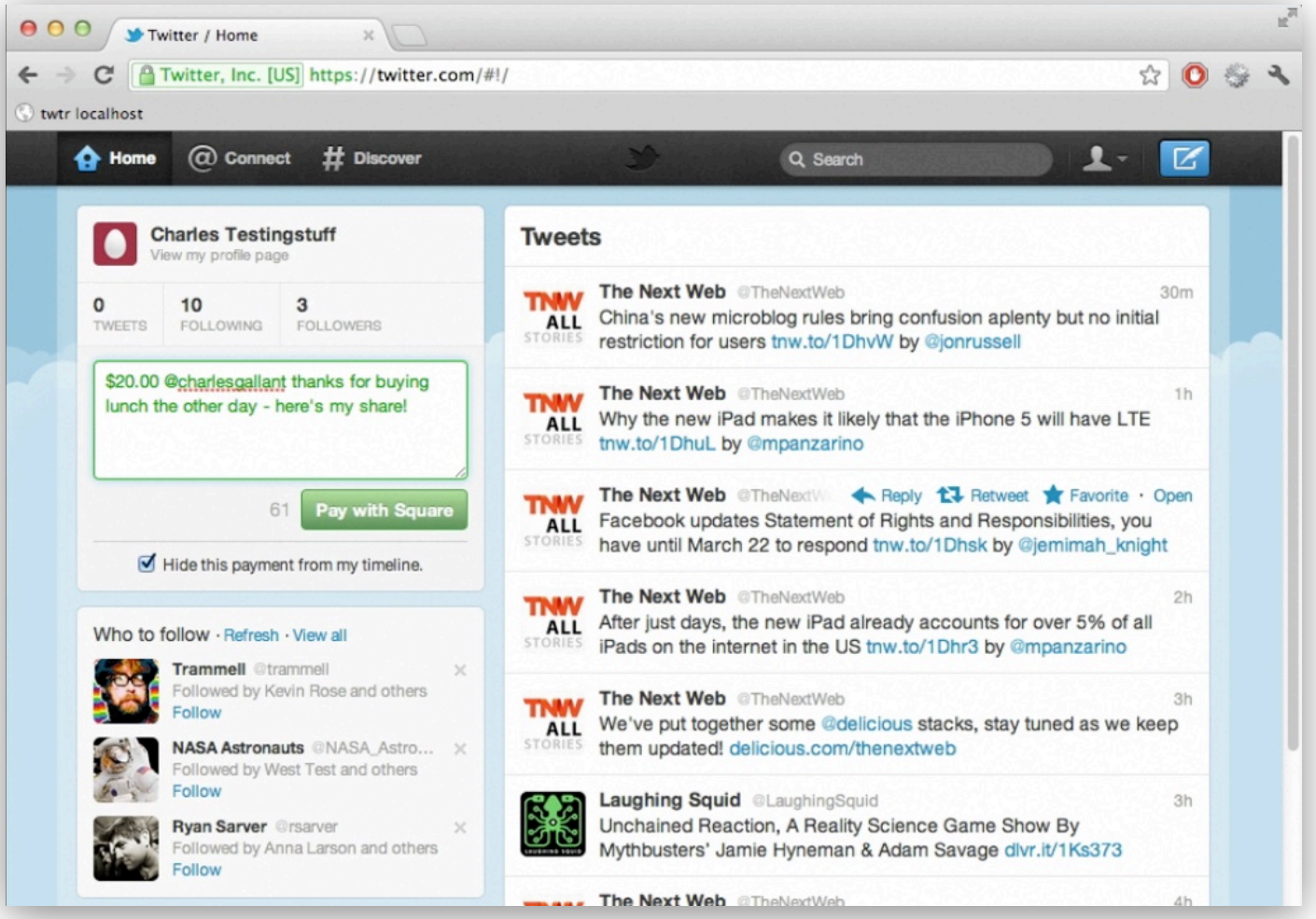


Example of a Twitter signup flow proposal (rough wireframes… one of 100+ iterations based on their 2013 design).



Our challenge with Twitter was to make the platform accessible; to help the “non-technically aligned” quickly understand and benefit from the service. It was 2013, after all… Twitter was still dominated by tech insiders. Creating and curating a timeline (via Following others) is a complicated first leap, for the average American in 2013.

Here, curious + potential users can be led into the experience by a family member, and witness a Timeline that they care about (and therefore value) before committing their personal info.



Many other prototypes and concepts were made at West. This one proposed a Square + Twitter integration as a way to pay with a tweet. As the user enters a specific series of characters (in this case, a $ followed by a few numbers, and a username), the CTA changes from Tweet to Pay with Square.

This was presented to the client via javascript DOM injection, and demo'd on Twitter.com in realtime. We brought a backup video to the meeting just in case Twitter's CSS selectors changed in the time it took us to travel to the Twitter office :).

Company: Poke New York

Role: Flash Developer / Creative Technologist

Poke was where I met some of the best web & mobile engineers in New York. During the boom of elaborate Flash “microsites,” we found opportunities to make groundbreaking digital experiences for some of the world’s top brands. While my role was primarily a web developer, it was an environment where we could pitch new business ideas, brainstorm campaign strategies, and explore what was possible with emerging mobile technologies.

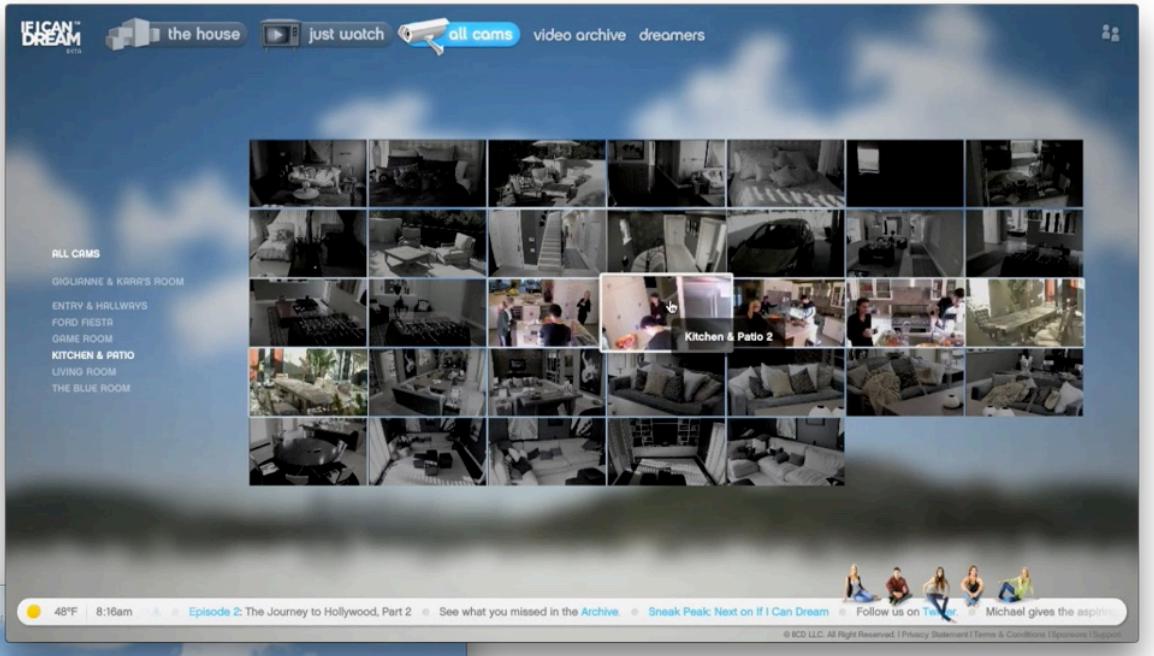
In retrospect, the death of Flash and the death of microsites was a wonderful thing: It forced me into new areas of technology (like Objective-C) and ushered in the era of responsive (and more responsible) design thinking.

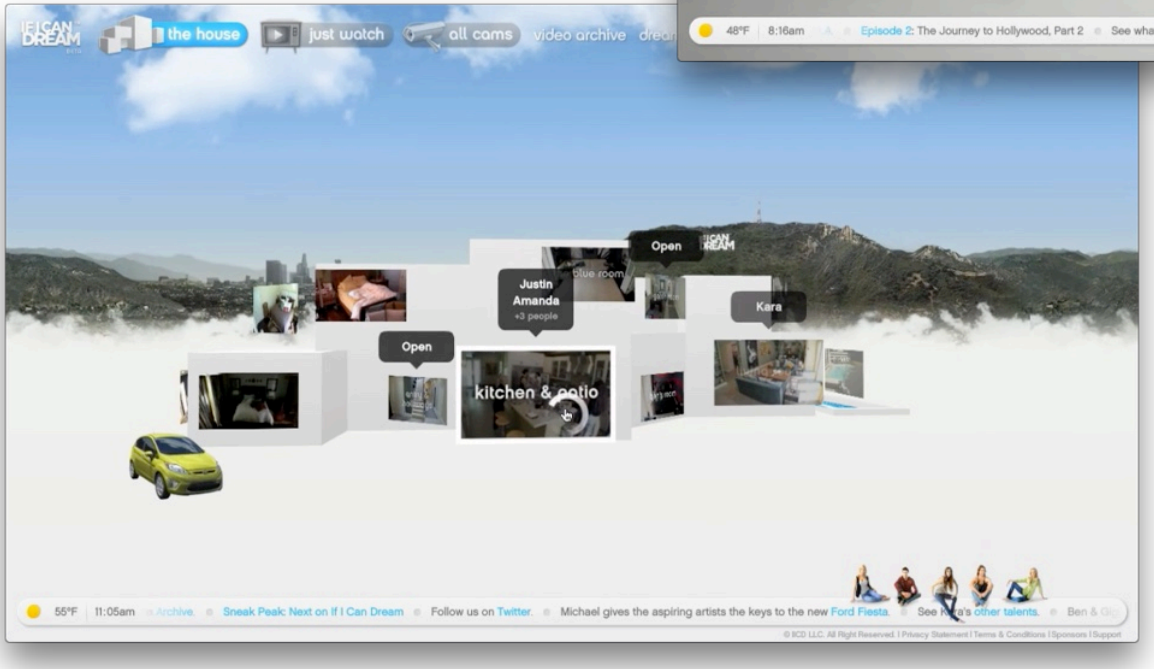
Warning: We’re going way back at this point… this work is ancient!

Agency: Poke New York
Role: Flash Developer
Year: 2007
Client: Teroforma

With this project, I was tasked to take a collection of hundreds of products (meticulously photographed in perspective space), and create a storefront that was both place setting and shopping cart. As users added products to their table, they would need to be *properly* arranged according to the rules of table setting etiquette, which meant I needed to develop a sorting & stacking algorithm.

Agency: Poke New York
Role: Flash Developer
Year: 2008
Client: 19 Entertainment





Our task was to push the limits of Flash & ActionScript, and create a website for a concept show called “If I can Dream.” With dozens of live cameras, a 3d-based UI, and enough real-time data to make a 2008 laptop overheat, it was one of the most ambitious websites of its time.

Context is everything, and the devil is in the details. While the examples herein aim to tell a broad overview, there's a lot more I could say about my journey.

Again, if you're receiving this portfolio, it means I would love to talk through these stories, and find a way to work together. I hope you decide to reach out :)

Until then, Cheers 

charlesgallant@gmail.com

linkedin.com/in/charlesgallant