goto;

GOTO
CHICAGO 2023
—
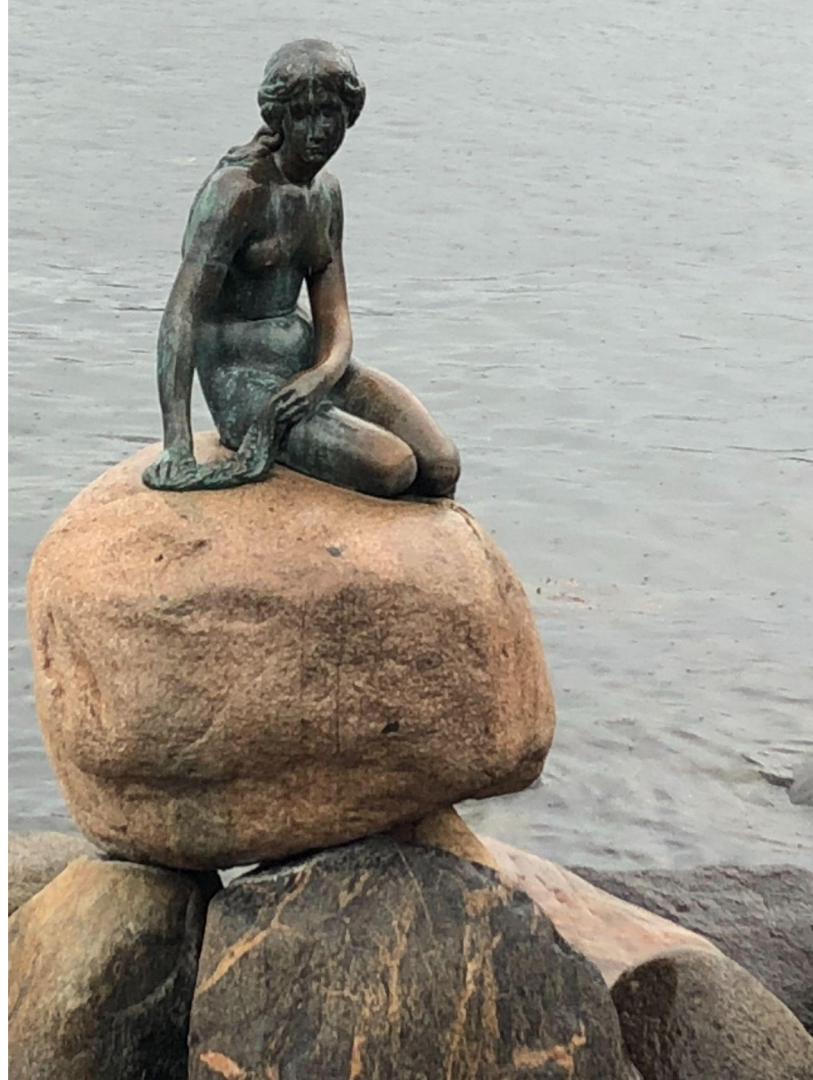
Run circles, not relays

#GOTOchgo

👋

I'm Gayle Silverman.

I'm passionate about communication & how we talk to each other.

I'm known for saying "two things can be true at one time."

You'll often hear me greeting people by saying "hi friends."

I'm a principal delivery lead at TXI (that means I work in product & execution).

It's a lot easier if we work together. A lot. The whole time.

Reject the idea that engineering, design, & product are destined to battle.

Fight the problem, not the people.

Roles on integrated teams
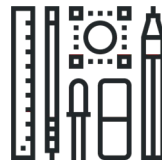
Product Strategists   **Engineers**   **Product Designers**   **Delivery Managers**

Don't get hung up on titles. Look for mindsets.

Engineers can be great at product. Designers can often code.

What really matters is a commitment to ongoing collaboration and transparent communication.

cross functional

≠

integrated

Cross functional teams, even agile ones,

often move forward via handoffs.

Progress can feel linear.

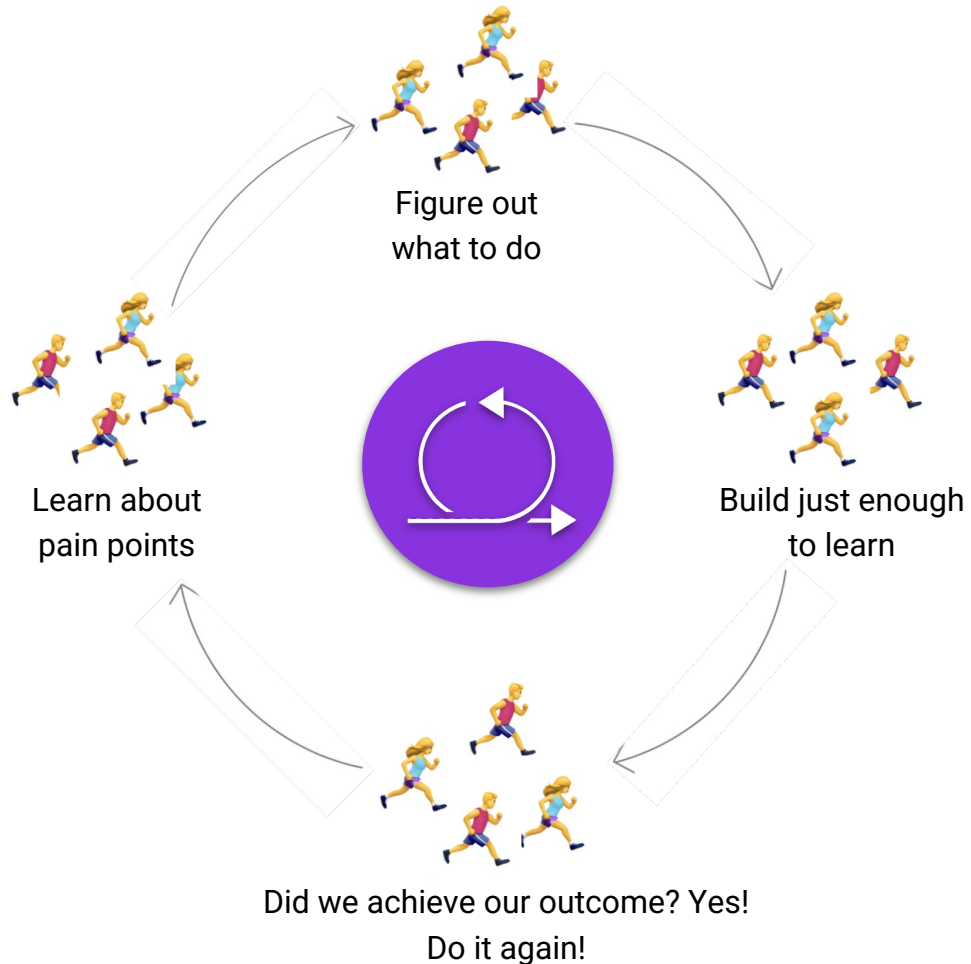Analysts gather requirements ⟶ Designers create wireframes ⟶ Engineers build software ⟶ We're done?

Integrated teams collaborate in each phase.

No siloed handoffs.

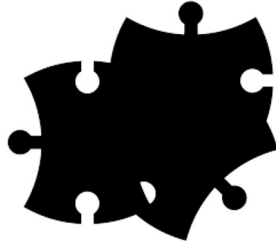Continuous, incremental shared progress.

Figure out what to do

Build just enough to learn

Did we achieve our outcome? Yes! Do it again!
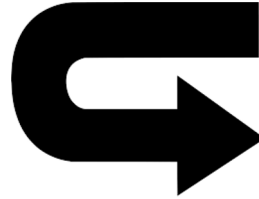
Learn about pain points

What happens when we don't work as integrated teams

Silos

Misalignment

Rework

Rage

Integrated teams operate collaboratively & collectively

〉 Everyone's north star is "how might we" solve this user's pain point

〉 Product identifies the desired outcome

〉 Design informs technical decisions 🔄 technical decisions inform design

〉 Develop shared language

〉 Ability to disagree and commit

〉 See something, say something mindset

What we gain as integrated teams

If we have

› Fewer siloes

› More representation in discussions

› Shared understanding

› Constant collaboration

Then we gain

› Improved delivery speed

› Reductions in misalignment and rework

› Ability to quickly respond to change

› Continuous growth

Integrated Team Principles

Integrated teams begin by defining what outcomes we're working towards

What user needs are we addressing?

What business goals are we advancing?

How will we measure success?

Embrace continuous cycles of discovery & delivery

〉 **Discover** enough to align on the problem to be solved

〉 **Define** the leanest version of what we'd need to validate our hypothesis

〉 **Build** just enough to test our hypothesis

〉 **Measure** to understand if what we've done gets us closer to the desired outcome

Integrated Team Practices

Discover

Empathize with our users pain points. Learn as much as we reasonably can. Align on a problem to solve.

- Design, engineering, product, and delivery **immersed in user research**.
- Identify the right **solvable challenges** from a feasibility, desirability, and viability perspective
- Plan for **technical discovery**
- Discuss if our **next big bet** should be technical or experiential

Define

Define

Move from discovery's preliminary insights and generative research, to product shaping and definition.

- **Everyone ideates** - product, design, engineering, delivery
- Opportunities evaluated based on **business value, design complexity, and technical complexity**
- Validate ideas from **user and technical** perspectives
- Prioritization informed by **all disciplines**
- Exploratory design **includes feedback** from all team members
- Foundational design optimizes for **reusable components**
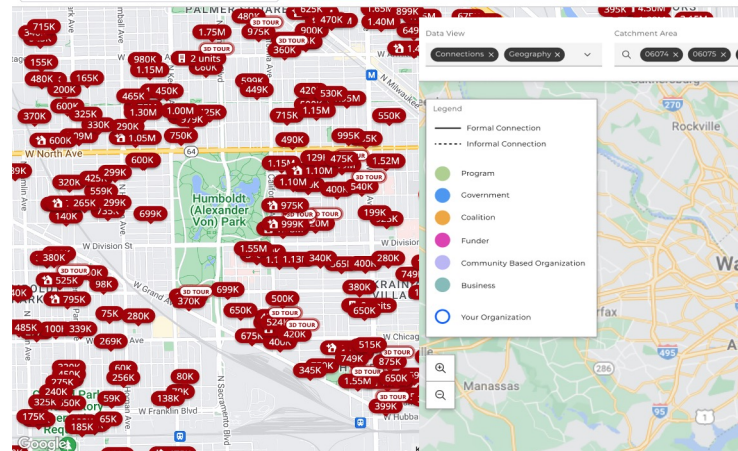
Build
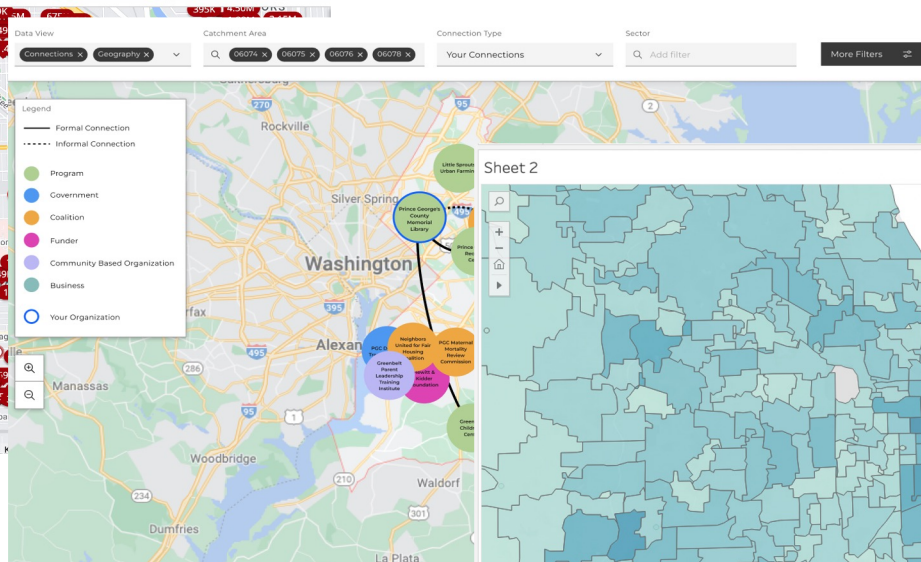
Inspiration from
discovery workshop

Concept testing
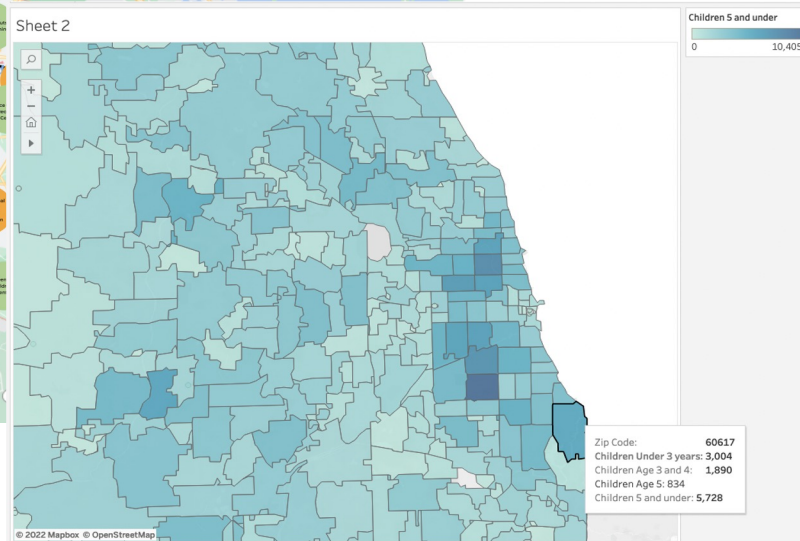
Technical prototype
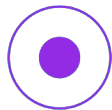
Build

It's time to build the thing! Get your product live and in the hands of end users to confirm we're making a real impact.

- Incorporate **cross-functional pairing** sessions
- Create a **culture of surfacing** unexpected design complexity
- Be prepared to **negotiate trade offs** once development begins
- Leverage **async reviews**
- Include **UX checks** as part of QA
- Embrace **ad hoc collaboration** for troubleshooting, fine tuning & polish

Measure

We want to reduce manual update of contact info

Product
What's the lowest friction, easiest user interaction to get us to that outcome?

Design
Is a visual dropdown a viable way forward? I found a library to help.

Engineering
Fits with our stack. Let's do it.

Measure

Learn how close we came to meeting our desired outcomes. Analytics, user feedback, and other metrics give us what we need to iterate on product and team success.

- Is drop off related to a **technical choice or design**?
- Would addressing tech debt **improve an experience** or do we need to re-envision the flow?
- Can we **solve the problem** a better way?

Discover

Getting Started

Build empathy for each other's needs & workloads

Three simple starters

Practice conversing constructively

Commit to working flexibly

Establish norms

Teams that communicate transparently, achieve collectively

⟩ Have strong points of view loosely held

⟩ Give actionable critique

⟩ Ask for what you (really) need

⟩ Radiate information consistently to people outside "the room"

⟩ Say no to requests without shame, judgement, or fear

⟩ Confidently raise an issue when you're the first person to spot it

That one time where product and engineering both insisted a common interaction pattern didn't exist.

Design, thanks for holding the course.

Optimize for the practice, team, AND the individual

〉 Decisions made after input from all disciplines (this is NOT the same as decision by committee)

〉 Make work visible

〉 Strive for equitable access of shared tools (Freehand, Zeplin, Miro)

〉 Don't block the box

〉 Nothing is untouchable

〉 Respect practice principles

Actual Fargo inspired Kanban board. Low tech, but very visible.

1   Stewardship

We are capable of better outcomes when we tap into the knowledge base we have as a collective and look to our engineers to contribute to definitional work, idea generation, and innovation efforts that exist outside of code alone.

2   Creativity

Because our engineers have direct contact with clients from the earliest stages of projects, our role is like that of an artist given a commission: we take fuzzy human expressions of need and turn them into tangible products. We deploy our coding skills to an immensely creative end.

3   Collaboration

We recognize that we are capable of better outcomes when we tap into the knowledge base we have as a collective and look to our engineers to contribute to definitional work, idea generation, and innovation efforts that exist outside of code alone.

4   Pragmatism

We pride ourselves on our business acumen and empathy for users. While we're often deep in code, we never lose sight of what our technical work is in service to.

5   Continuous improvement

We actively seek out new information and mine analytics to understand and improve user interactions. We also strive for a growth mindset, monitoring tech trends and emerging best practices.

Be clear on what makes each person thrive

〉 Create a **team agreement** before beginning a new project (example on next slide)

〉 **Agree on a decider** before decisions need to be made

〉 Explicitly **state pairing** styles

〉 Discuss when people do their **deepest work**

〉 Embrace people's **work love language**

Establish team norms

Question	Your Response
Pronouns What are your pronouns?	she / her
Core Working Hours / Time Zone 🖥️ What hours are you working out your core? JK, what hours can we expect to find you online? e.g. 8a-5p CT	8 - 4:30 CT
Off-Limit Hours 🚫 Do you have any protected time or time that is you'd like the team to try to respect as "off limits"? e.g. lunchtime @ 12p CT	Prefer no meetings after 3PM CT
Communication Preferences What's the best way communicate with you? Do you prefer @ mention or just your name in Slack? How often do you check email?	Slack with an @ mention
Async Communication How do you feel about async communication in the absence of meetings? Love it? Leave it? Depends on the situation?	Love it / If we have more than 2-3 exchanges on a topic, let's jump to zoom

Sample team agreement

Feedback 🗣️ How would you like to receive feedback?	As close to the moment as possible / In whatever way is easiest for you to share
Growth / Support 🌱 What are some ways we can support your growth? Your individual workstyle?	Product strategy
Values / Motivation ✨ What lights you up / motivates you?	Feeling like I'm contributing to the team / a kudos now and then is nice / Getting actionable feedback
Count on me for... 🙌 Is there anything you'd like to call out? e.g. honest feedback, pairing, a good joke	Helping untangle people / Framing hard conversations
Favorite Emoji 🧡 What's your favorite emoji right now?	lol sob 😂 (it's not in Miro)
What are you a silo for? What siloed context have you been holding on to that we need to discuss as we start building this latest version of our team?	I'm an empty silo

Understand how to effectively ask each other questions

If you think

〉 A dropdown will be **easier**

〉 This is **harder** than it needs to be

〉 This is too **complex**

〉 There's an **unaddressed** pain point

〉 This is **unnecessary**

Try asking

〉 How **important** is this to the interaction?

〉 Can we replace this with an existing **pattern**?

〉 What's the **value** of this interaction?

〉 What's the easiest way to **validate** my hunch?

〉 **How often** do we think this interaction will happen?

Watch out for implicit red flags

Becoming an integrated team takes time, commitment, and practice. Be on the lookout for behavior that is antithetical to collaboration.

> Deviating from collaboration norms: skipping meetings, blocking decisions last minute

> Hoarding information

> Privileging professional titles over individual suggestions

> Resistance to negotiation

> Decision made via side conversations

Learn more

〉 Reach out to Gayle with any follow up questions
gayle[@]txidigital[dot]com

〉 Read <u>TXI's article</u> on integrated teams

〉 Check out Continuous Discovery by Teresa Torres

〉 Sign up for my newsletter series on Pragmatic Communication (QR code on the right)

〉 Read Kara Carrell's <u>pre-PAIR-ation Checklist for Pairing</u>

TL; DR
Do all the things.
Together.

goto;

Don't forget to
vote for this session
in the **GOTO Guide app**