

QRC Lean UX

Based on Lean UX – Designing Great Products with Agile Teams. Aug.'2017, Henny Portman

Principles

Principles to guide team organization are:

- Cross-functional teams
- Small, dedicated, co-located
- Self-sufficient and empowered
- Problem-focused team

Principles to guide culture:

- Moving from doubt to certainty
- Outcomes, not output
- Removing waste
- Shared understanding
- No rock stars, gurus, or ninjas
- Permission to fail

Principles to guide process are:

- Work in small batches to mitigate risk
- Continuous discovery
- GOOB (getting out of the building): the new user-centricity
- Externalizing your work
- Making over analysis
- Getting out of the deliverables business

RESEARCH & LEARNING

Feedback and research

- Collaborative discovery: as a team review, decide who to speak, create interview guide, break your team into research pairs, arm each pair with a version of the MVP, meet the customer, interview and take notes, begin with questions, conversations, and observations, demonstrate the MVP, collect notes and customer feedback, switch roles
- Continuous learning: three users every Thursday, simplify your test environment, making sense of the research
- Monitoring techniques: customer service, on-site feedback surveys

OUTCOMES, ASSUMPTIONS, HYPOTHESES

Driving vision with outcome

- Project's problem statement
- Declare assumptions (4 types: Business outcomes, Users, User outcomes and Features)
- Transform assumptions into hypotheses (tactical and testable)

DESIGN IT

Collaborative design

- Build a shared understanding, generate and converge ideas by using:
- Design Studio: problem definition and constraints, individual idea generation (diverge), presentation and critique, iterate and refine in pairs (emerge), team idea generation (converge)
 - Design systems, style guides, collaborative design sessions, and simple conversations

Lean UX Process

CREATE AN MVP

Minimum Viable Products and Prototypes

MVP: the smallest thing you can make to learn whether your hypothesis is valid.

- Creating an MVP to understand value: get to the point, use a clear call to action, prioritize ruthlessly, stay agile, don't reinvent the wheel, measure behavior
- Create an MVP to understand implementation: be functional, integrate with existing analytics, be consistent with the rest of the application
- Final guidelines for creating MVPs: it's not easy to be pure, be clear about your learning goals, go small, you don't necessarily need code, the truth curve
- Examples of MVPs: landing page test, feature fake (aka button to nowhere), Wizard of Oz
- Prototyping: paper, low-fidelity on-screen mockups, middle- and high-fidelity on-screen prototypes, coded and live-data prototypes.