goto; conference aarhus

FAST, EASY USABILITY TRICKS FOR BIG PRODUCT IMPROVEMENTS

Chris Nodder

Chris Nodder Consulting LLC

INTERNATIONAL SOFTWARE DEVELOPMENT CONFERENCE

gotocon.com

We already listen to our users…

Surveys: good at reinforcing your biases
Metrics: tell you *what*, but not *why*
Focus groups: tell you ???

Guided observation: find the pain points

"Get out of the building"

5 easy steps to a better product

1. Find some users to watch
2. Interpret what they tell you without bias
3. Create actionable product ideas
4. Turn your ideas into designs
5. User test your designs

…all before you even start coding!

Why?

- Quick way to improve the product
- Great team bonding experience
- Cheap
- Easy to interpret results
- Long-lasting value
- Stops arguments – you have real data

1. Find some users to watch

- Work out who you care about
  - If you say "everyone," you don't have good product definition (you have bigger problems)
- Seek them out in their environment

$ Advert on your site/mailing list/social media
$ Ask your sales people very nicely
$ Classified advert (e.g. Craigslist)
$ Post to social media/message board/club site
$ Ask friends and family to suggest
$ Go to where these people hang out, grab them
$$ Advert on a suitable site
$$$ Pay a recruiter

Field observation

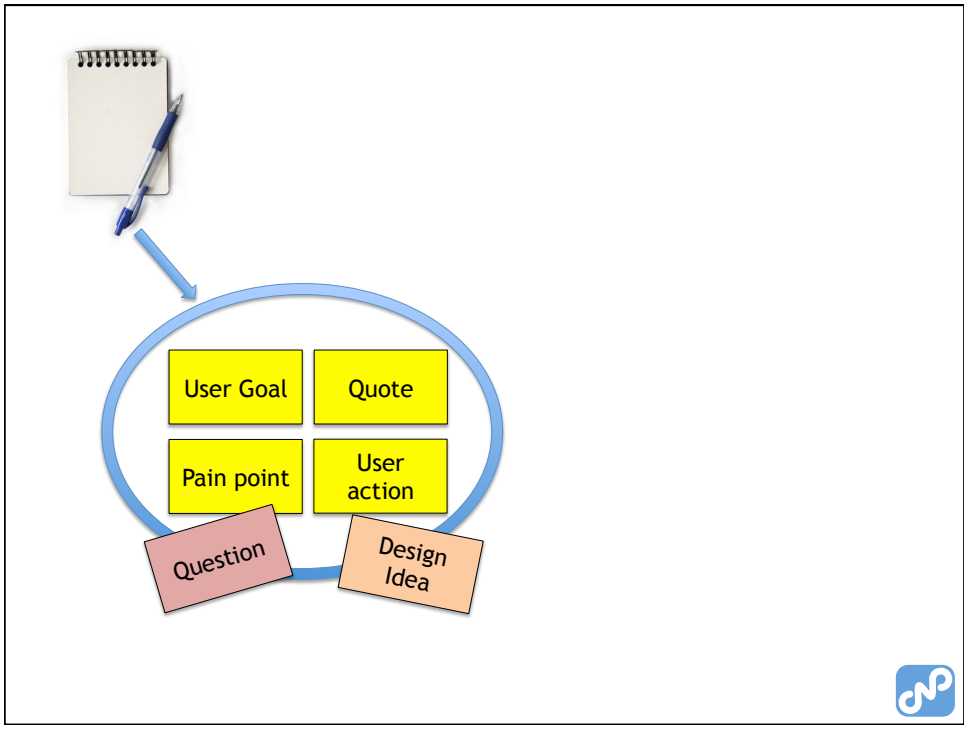
- Go to where your users are when they do the things you care about
  - Visit at the time they normally do the task, or ask them to save it up for you
- Primarily watch, don't speak
  - After the introductions, just be quiet
  - Sit behind/to the side of the user so you can see but aren't in the way
- Only ask questions to clarify
  - "Can you tell me more about …"
  - Don't make assumptions about the perceived cause of problems
- Take lots of hand written notes
  - Video is cool, but you'll never transcribe it and it can scare people
  - Take a couple of photos of the environment if you are allowed
- As many visits as you have time for
  - At least three visits per user type, at least five visits total
  - Each visit is normally 2 hours long (even if the task is shorter or longer)
  - Two people from your team on each visit (navigation, safety, note taking)

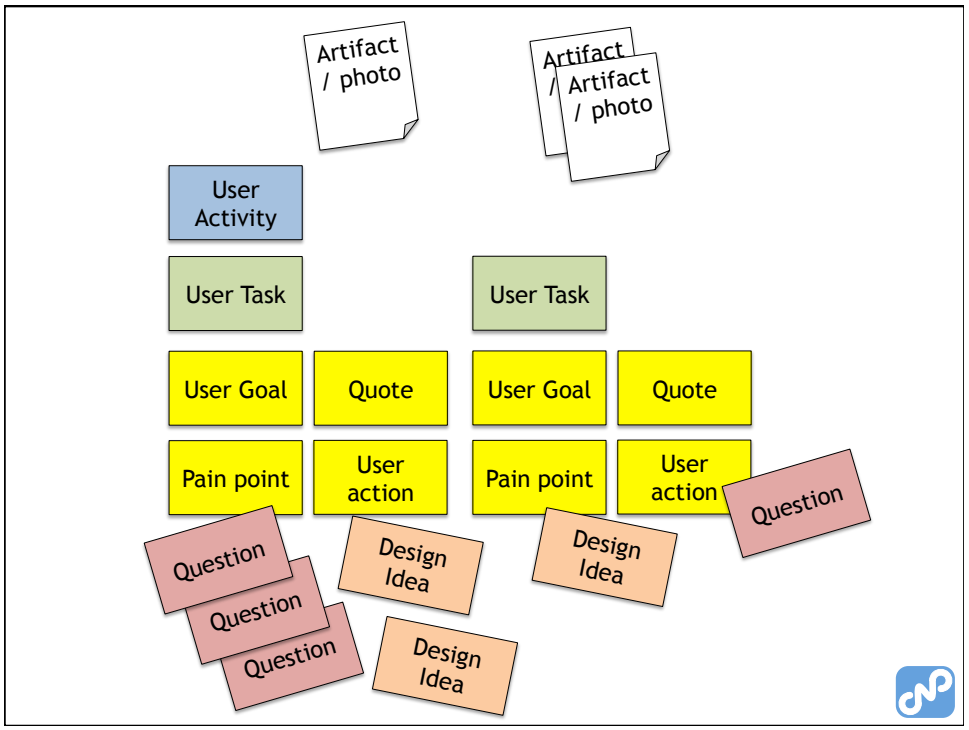
Field observation

- DO:
  - Take notes
  - Engage (smile)
  - Ask open-ended questions
  - Ask for examples (times when “it” happened)
- DON’T:
  - Engage in conversation
  - Sell them on your cool product idea
  - Ask them to predict the future

2. Create an experience map

- Every observation goes on a sticky note
- Every sticky note goes on the wall
- Sticky notes are grouped into tasks
- Tasks are arranged chronologically
- If you think of design ideas, add them on a different colored sticky note
- If you think of more questions, add them on a different colored sticky note


User Goal
Quote
Pain point
User action
Question
Design Idea


Artifact / photo
Artifact / photo
User Activity
User Task
User Task
User Goal
Quote
User Goal
Quote
Pain point
User action
Pain point
User action
Question
Question
Design Idea
Design Idea
Question
Question
Design Idea

Experience Mapping

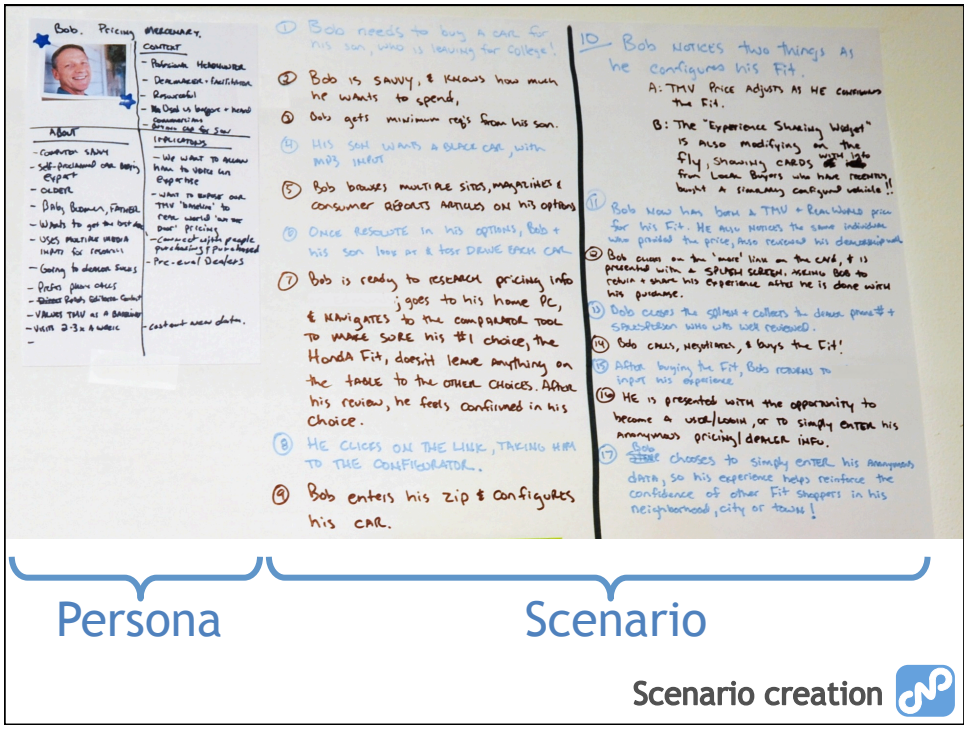
Successful experience maps

- Think of making the map as a data party
- Bring everyone who was on visits into the room at the same time
- Everyone writes and places their stickies, mainly without comment
- Welcome disagreements – they highlight where the interesting stuff is
- Focus on user pain points (resolving those in your product is a big win)
- Information radiator: Put the finished map in a busy place
- Bias is reduced by multiple observations, and by focusing on the problem not the solution

3. Create actionable product ideas

- You need to get the messy map into a manageable form
- What areas will you focus on in your product?
- Write scenarios that cover the areas users talked about in site visits ("User Activities" in your experience map)
  - Create descriptions of how the people you visited could use your (new) product to solve their problems*
- Don't describe specific UI yet… that's the next step. Instead, describe behaviors and outcomes

*These are similar to use cases, but I prefer to use real people or personas as "the user"



Scenario creation

4. Turn your ideas into designs

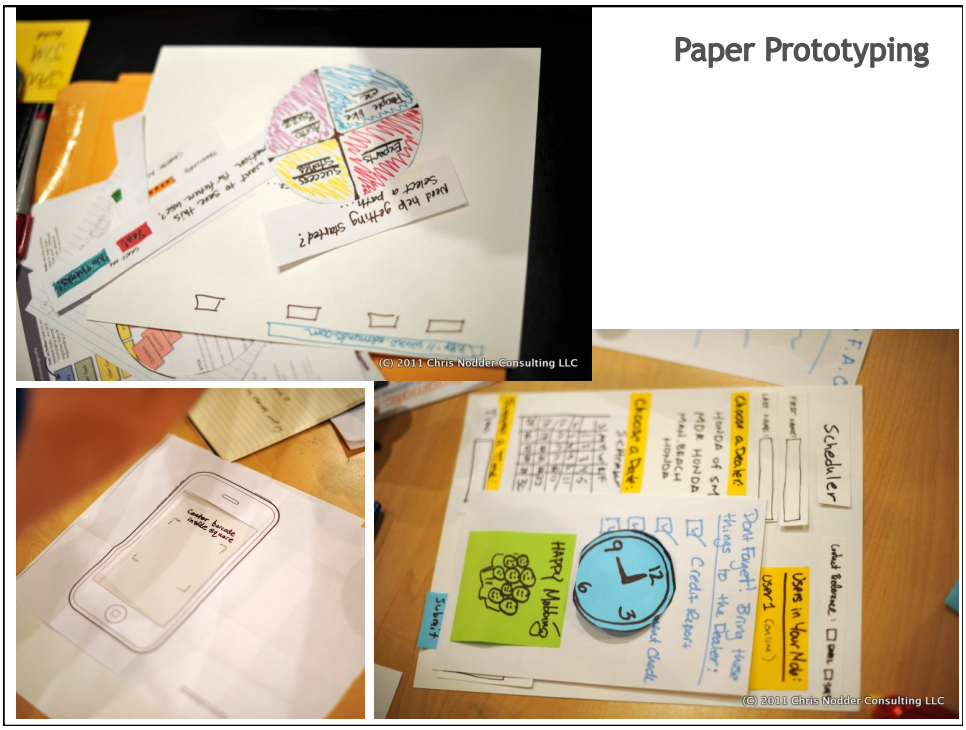
- Hold a design charrette!
  - Choose one scenario and get every team member to sketch a design solution
  - The sketch can be UI, a comic book/storyboard, or anything else that gets an interaction idea across. Artistic ability isn't important
  - Each individual presents their sketch to the group
  - All sketches are pinned on the wall
  - Everyone "dot votes" the concepts they like
- Do another round, or a smaller group takes all the good ideas and creates a new summary sketch
- If you have a UI expert on the team, you can use this as a critique session to teach good basic UI principles as well (for instance Nielsen's 10 heuristics*)

* www.useit.com/papers/heuristic/heuristic_list.html

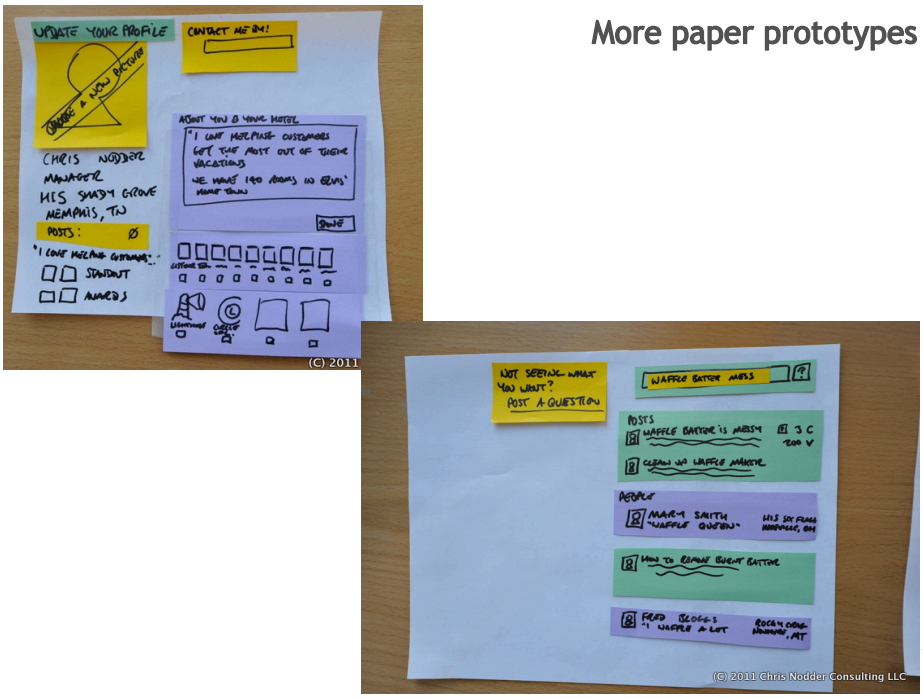
Design charrette

Build a paper prototype

- The charrette left you with some UI sketches. Now you have to make them real enough to "work"
- Build only the interface elements needed to enable the scenario you wrote from field observation
  - This way you avoid feature creep and create the minimum viable product
- Have one person read the scenario out loud while another works through the UI
  - Because the scenario describes behaviors and outcomes, it should be easy to see if the UI meets the criteria
- Tip: Create each UI element on a separate piece of paper so that you can rearrange them or remove them without re-drawing everything



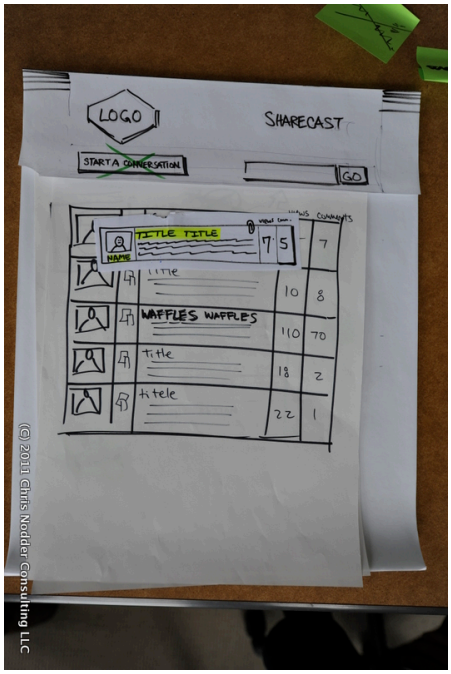
More paper prototypes

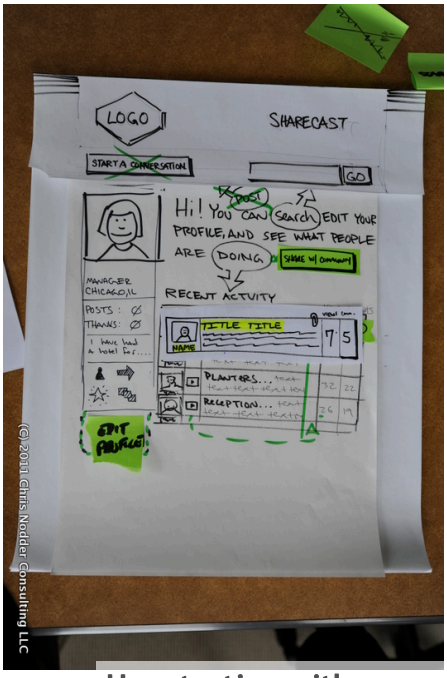


5. Run a user test*

- Use your paper prototype
  - You don't need code to run a user test. It's amazing how much you can mock up with paper.
- Warm bodies
  - Your users don't have to be very representative for early tests – the interaction ideas should be understandable by most experienced computer users
  - About 5 users is enough to be sure that the problems you see are real
- Tasks
  - Use your scenarios to write tasks for users to perform
  - Make sure that the wording of tasks doesn't give away the answer
- Observers
  - Team members watch (remember the duct tape)
  - They write down observations. Save "solutions" until after the sessions
- Reward
  - Find something to say thank you to participants. Movie tickets, marketing give-aways, etc.
- Output
  - A list of issues with the prototype. Fix them (and re-test if necessary) before coding

**Users test the prototype, you don't test users*





User testing with paper

Now you can start writing code

- It's crazy that people consider starting to write code before knowing what to build
- Developers go on visits, create the experience map, participate in charrettes, watch studies – everyone should be too busy to write code!
- If developers get itchy fingers, make them do back-end work, not UI work

Recap: 5 easy steps

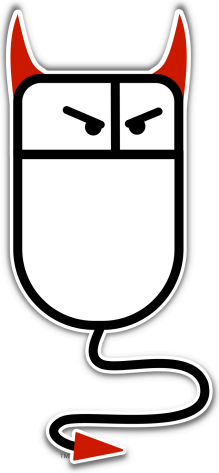
1. Find some users to watch
2. Interpret what they tell you without bias
3. Create actionable product ideas
4. Turn your ideas into designs
5. User test your designs

…all before you even start coding!

Benefits

- Great team bonding experience
- Cheap
- Easy to interpret results
- Fast way to improve the product
- Long-lasting value
- Stops arguments – you have real data

chris@nodder.com
@uxgrump



Taming evil interfaces
Creating heavenly experiences