How to Reduce the Graveyard of Software Tools

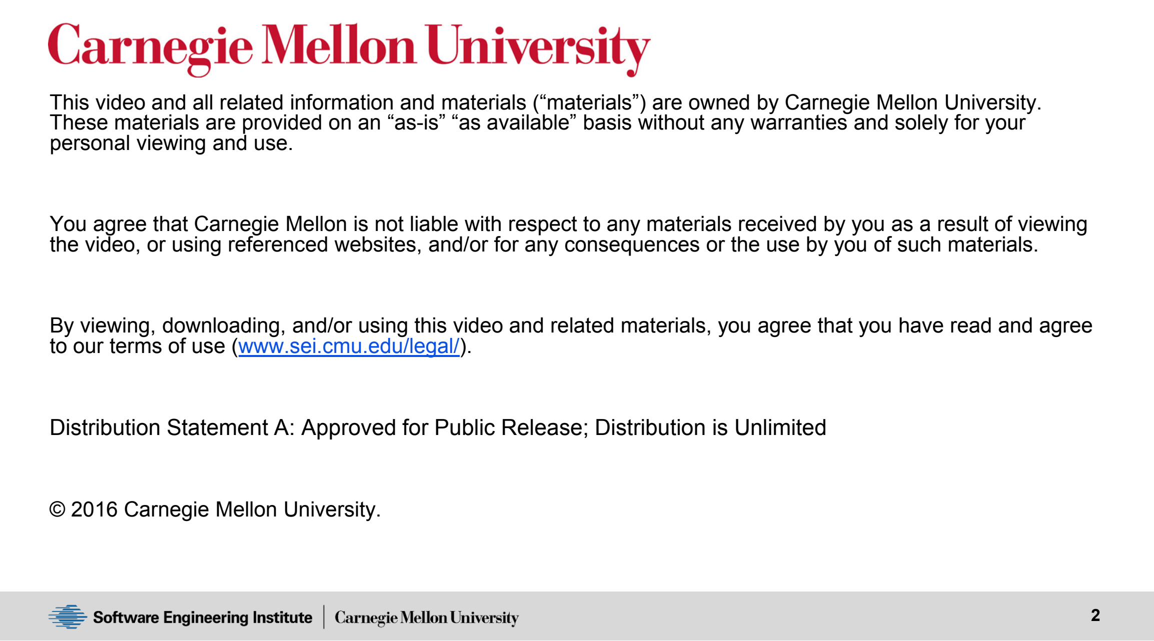
Table of Contents

SEI WEBINAR SERIES | Keeping you informed of the latest solutions

SEI WEBINAR SERIES | Keeping you informed of the latest solutions

Software Engineering Institute | Carnegie Mellon University

Carnegie Mellon University



Carnegie Mellon University

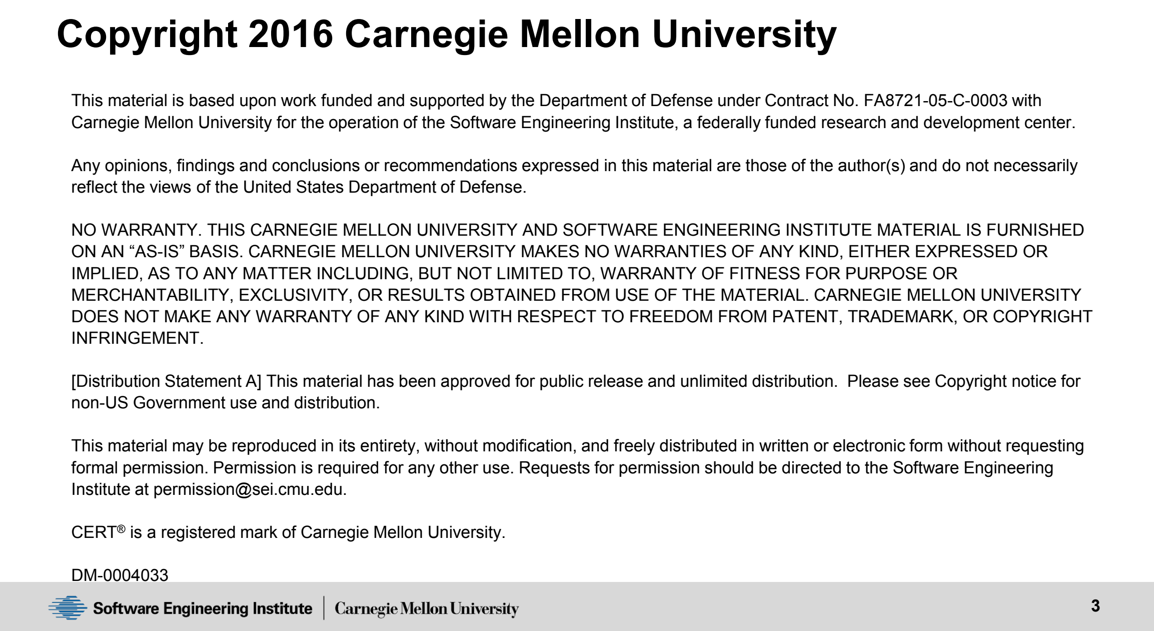
This video and all related information and materials ("materials") are owned by Carnegie Mellon University. These materials are provided on an "as-is" "as available" basis without any warranties and solely for your personal viewing and use.

You agree that Carnegie Mellon is not liable with respect to any materials received by you as a result of viewing the video, or using referenced websites, and/or for any consequences or the use by you of such materials.

By viewing, downloading, and/or using this video and related materials, you agree that you have read and agree to our terms of use (www.sei.cmu.edu/legal/).

Distribution Statement A: Approved for Public Release; Distribution is Unlimited

Copyright 2016 Carnegie Mellon University



Copyright 2016 Carnegie Mellon University

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Department of Defense.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN “AS-IS” BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

CERT® is a registered mark of Carnegie Mellon University.

DM-0004033

Software Engineering Institute | Carnegie Mellon University 3

How to Reduce the Graveyard of Software Tools with UI/UX Capability

How to Reduce the Graveyard of Software Tools with UI/UX Capability

Jennifer Cowley

Mike Szegedy

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Software Engineering Institute | Carnegie Mellon University

© 2016 Carnegie Mellon University
[Insert Distribution Statement Here]

**004 Presenter: And hello from the campus of Carnegie Mellon University in Pittsburgh, Pennsylvania. We welcome you to the Software Engineering Institute's webinar series. Our presentation today is How to Reduce the Graveyard of Software Tools with UI/UX Capability. Depending on your location, we wish you a good morning, a good afternoon, or a good evening.

My name is Shane McGraw. I'll be your moderator for today's presentation. And I'd like to thank you for attending. We want to make today as interactive as possible. So, we will address questions throughout the presentation and again at the end of the presentation. You can submit your questions to our event staff at any time by using the ask a question tab on your control panel, or log into the chat tab on your control panel and post there.

We will ask a few polling questions throughout the presentation. And they will appear as a pop up window on your screen. The first question we'd like to ask is, "How did you hear about today's event?" And that will be live now.

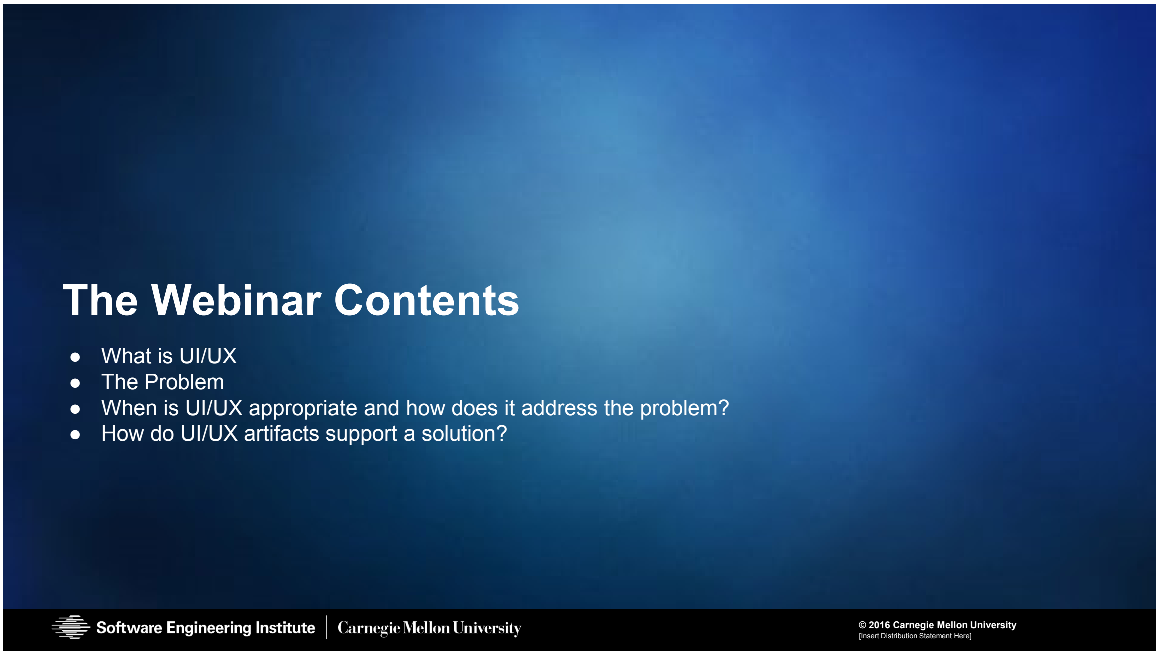
Another three tabs I'd like to point out are the download materials, Twitter, and survey tabs. The download materials tab has a PDF copy of today's presentation slides there now. And for those of you using Twitter, you want to be sure to follow @cert_division and use the hashtag seiwebinar. And at the survey tab, we ask that you fill out, upon exiting today's webinar as your feedback is always greatly appreciated.

And now, I'd like to introduce our two speakers for today. Dr. Jennifer Cowley is a principal researcher in the CERT division of the SEI. And prior to the SEI, during her tenure at North Carolina State University in pursuit of a doctoral degree, she managed a research lab, taught human factor courses, and was a full-time user interface designer at the SAS Institute and also worked on an internship at the MITRE Corp.

Mike Szegedy has been working in the field of user experience and design for over thirteen years. Working with the Human Computer Interaction Institute, he has done deep dive research into the following topics, trust building in aged populations online, contextual notifications on mobile devices, and educational development for underprivileged children. And he is currently a UX designer and researcher within our CERT division. And now, I'm going to turn it over to Jennifer Cowley. Jennifer, welcome. All yours, Mike.

Presenter: Well, thank you for the gracious introductions. We'd like to start talking a bit about how to reduce the graveyard of software tools. And I think that the concept of what's a graveyard is a fair question from the audience. And so, I just wanted to quickly make a point about that.

The Webinar Contents



The Webinar Contents

- What is UI/UX
- The Problem
- When is UI/UX appropriate and how does it address the problem?
- How do UI/UX artifacts support a solution?

Software Engineering Institute | Carnegie Mellon University

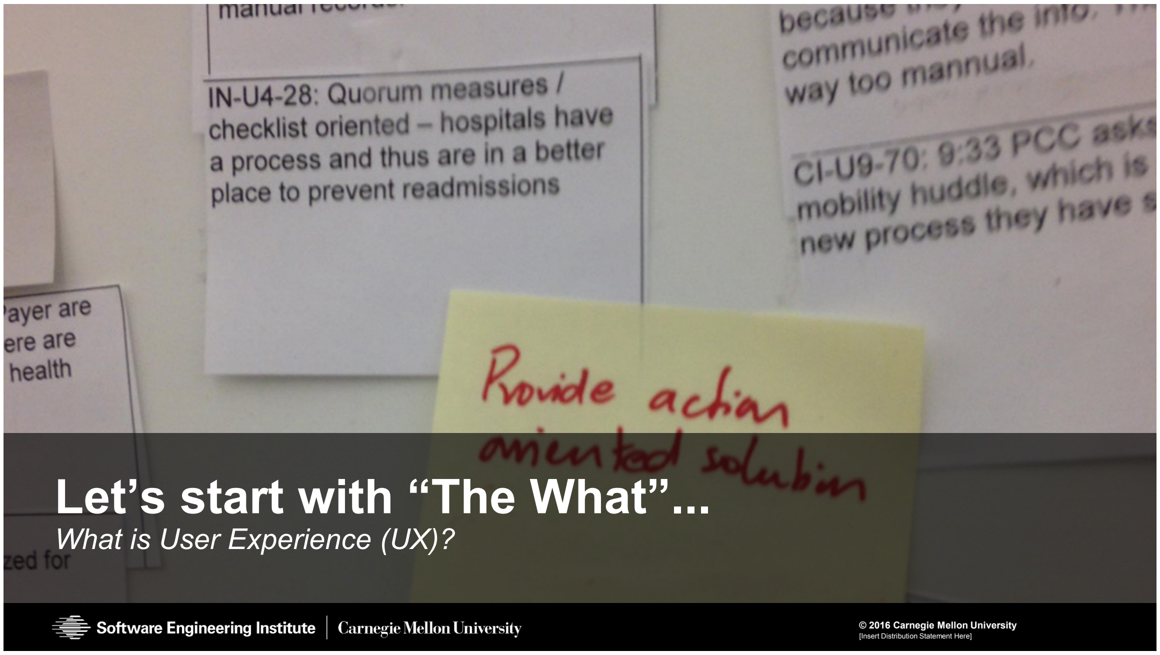
© 2016 Carnegie Mellon University
[Insert Distribution Statement Here]

**005 And what do I mean about a graveyard? Well, often we have failures of software tools that have been released in to the wild. And they don't perform, people don't use them, etc. So, in general when I talk about failures that lead to a graveyard of software tools, I'm just talking about software and hardware that we've developed that just fails to meet the tasking and the work of the user.

And so, I'd like to just quickly look at the webinar contents. And I wanted to take a step back and say that this is a little bit more of an intimate crowd. And we have much more freedom to accept and answer some questions from the audience. And we'd be happy to do that. So, we're not slated to hurry up and get through all of our slides on time. So, if there's a pressing question, please feel free to go ahead and chat. And we'll do our best to address them in the meantime. We're going to talk about the UI/UX, what really is it? And, go ahead.

Presenter: Yeah, so today we're going to be touching on a couple of different topics focusing on what is UI/UX. There seems to be a little bit of confusion in the marketplace and in different organizations. Addressing the problem directly, what is causing these software products to not be adopted, to fall into disuse? When is UI and UX appropriate? When does it fit into your timeline? And how do the different artifacts that come out of the process support your solution?

Let's start with "The What "...


IN-U4-28: Quorum measures / checklist oriented – hospitals have a process and thus are in a better place to prevent readmissions
because they communicate the info. way too mannual.
CI-U9-70: 9:33 PCC asks mobility huddle, which is new process they have
ayer are ere are health
zed for
Provide action oriented solution

Let's start with "The What"...

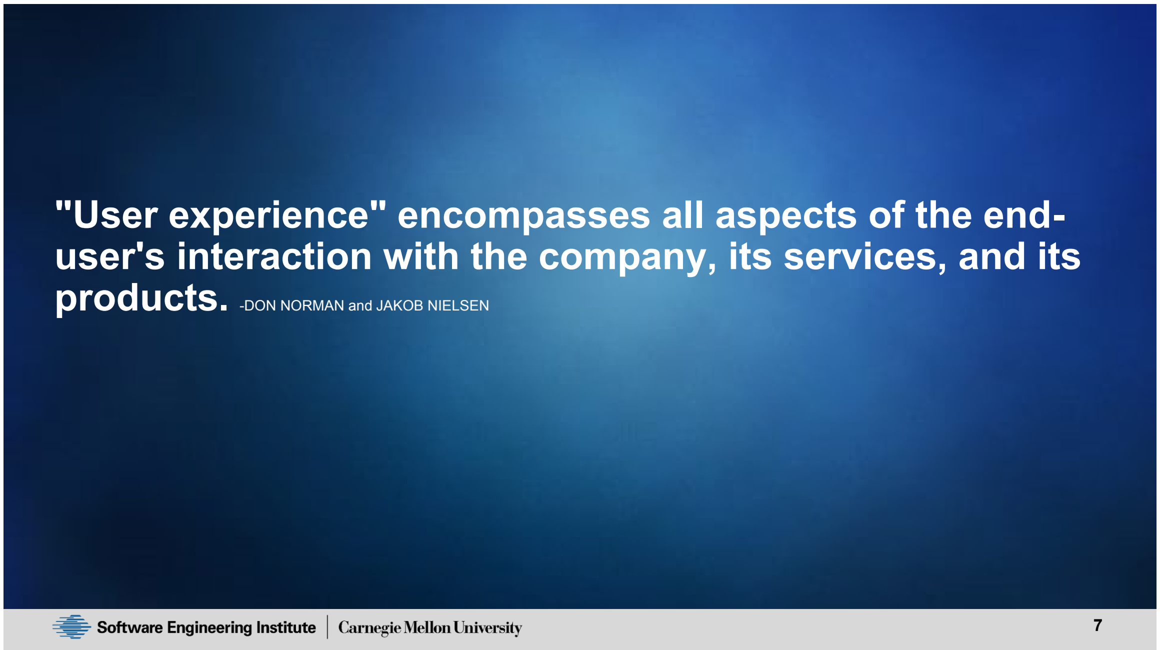
What is User Experience (UX)?

Software Engineering Institute | Carnegie Mellon University

© 2016 Carnegie Mellon University
[Insert Distribution Statement Here]

**006 So, the first thing I want to start with is the what, which is what is user experience.

"User experience" encompasses all aspects of the end-user's interaction with the company, its services, and its products. -DON NORMAN and JAKOB NIELSEN



"User experience" encompasses all aspects of the end-user's interaction with the company, its services, and its products. -DON NORMAN and JAKOB NIELSEN

Software Engineering Institute | Carnegie Mellon University 7

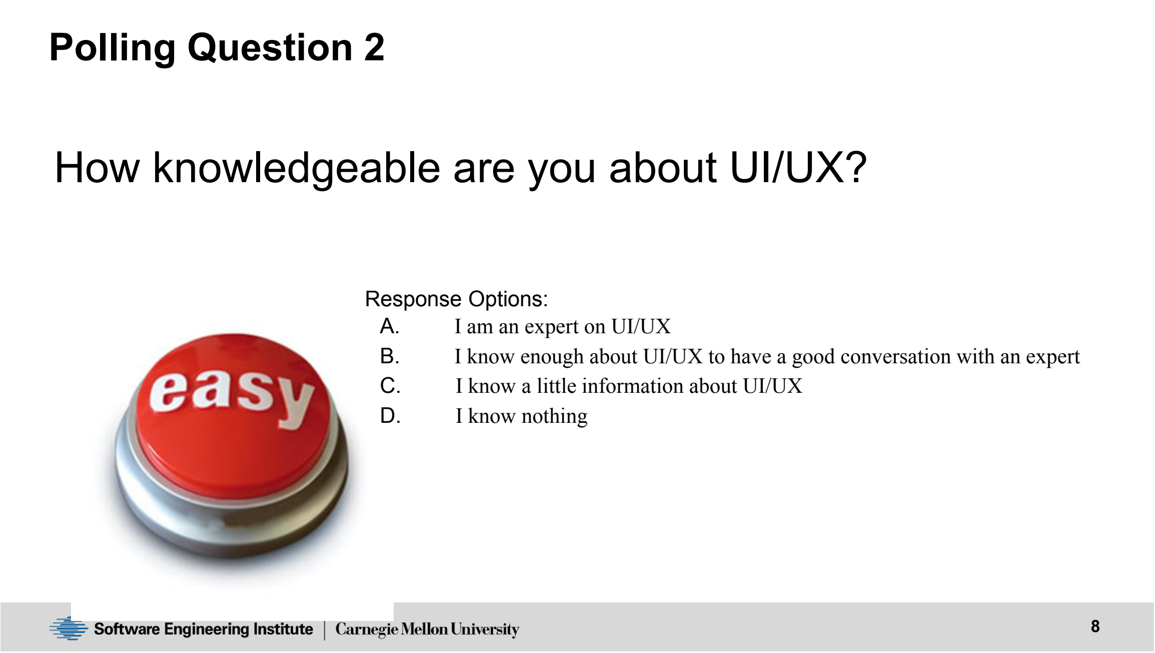
**007 Don Norman and Jakob Nielsen define user experience as encompassing all aspects of the end-user's interactions with the company, its services, and its products. And I think that that's a really compelling description because it covers the whole multifaceted environment that we find ourselves in in today's software and hardware environments.

Presenter: And I want to echo that in the sense that this definition of what a user experience professional is is really not about designing an interface exclusively. It's a much broader concept. So, we're looking at how well the tool fits with that user's capability in the work environment with which they find themselves. And remember that tools are often a part of a system. So, we look often at the system design.

Another point I wanted to make out was that user experience, we're going to use that generically and colloquially today. But what we mean by user experience could also interchangeably used with the term usability expert, usability evaluator, usability engineer, UI/UX designer/evaluator. So, there's a plethora of different titles and job titles that sort of fit under this umbrella of what we're just going to call user experience today. So, we'll just abbreviate that with UX.

Presenter: Yeah, that's a good point. Everything that we're going to be talking about today is going to be under the umbrella of generalities. Theoretically, in the future, we can explore very specific topics and concepts if the audience has interest in that. But today, we're going to try to cover the overall umbrella.

Polling Question 2



Polling Question 2

How knowledgeable are you about UI/UX?

Response Options:

A. I am an expert on UI/UX
B. I know enough about UI/UX to have a good conversation with an expert
C. I know a little information about UI/UX
D. I know nothing

Software Engineering Institute | Carnegie Mellon University 8

**008 Presenter: That's right. That's right. We don't want to talk over our audience or talk under our audience. So, I'm going to turn the question over to Shane.

Presenter: Right. So, we're going to pose our second polling question just to get an idea of your level of knowledge about UI/UX, expert, I know enough, I know little, or I know nothing. So, we'll give you about ten or fifteen seconds a vote. And that's going to give Jennifer and Mike a chance to kind of-- for their talking points, where to take the discussion. So, back to you guys.

Presenter: We really hate to talk above the audience and talk below the audience. We really want to make sure the audience is drawn into the conversation as much as possible.

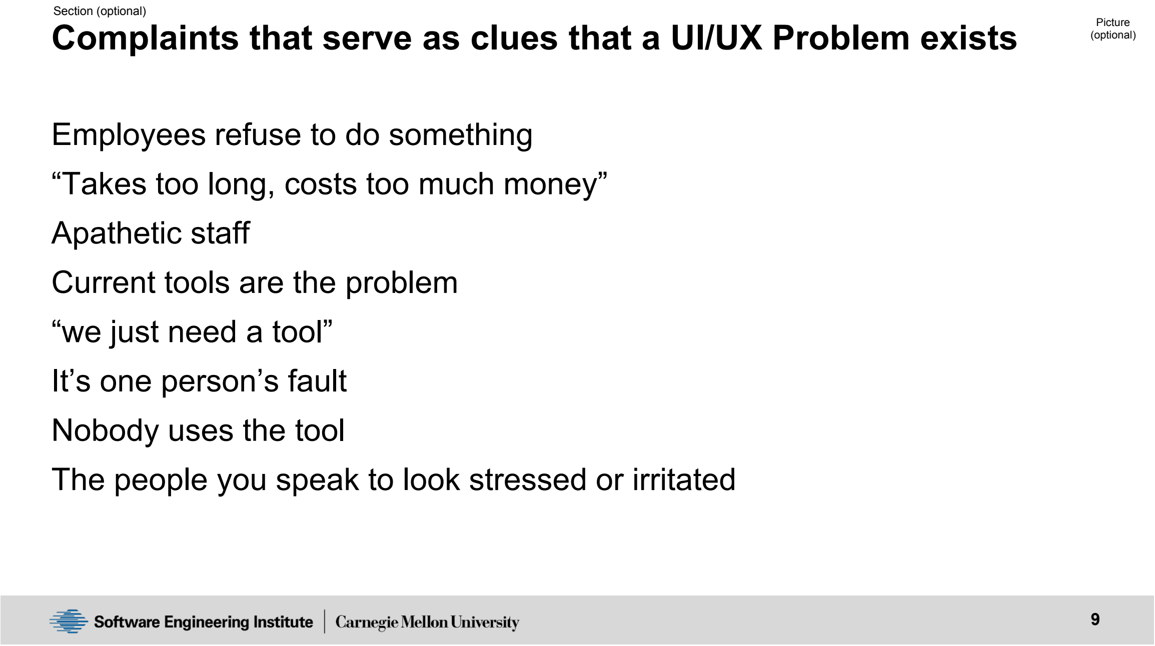
Presenter: Yeah, and I think that that's important. Oftentimes, when you encounter user experience designers in the marketplace, there are a lot of different backgrounds that they've come into the market from. I personally have a fine arts background. And then I went in for my masters and learned a lot of the experience that I've had through on the job training. But people come from many varied backgrounds. And so, their definition of what user experience is to them oftentimes is very different. Some places, it's very visually based. Other places the psychology kind of takes over.

Presenter: That's right. That's right.

Presenter: So, I'm going to chime in with those results. We've got fifteen percent at the expert level, forty-four percent I know enough to have a good conversation, thirty-seven percent I know little, and four percent I know nothing. So, hopefully that's helpful. It's a pretty big mix but--

Presenter: Oh, that's right. That's right. So, sort of in the middle. And we'll do our best to address people's concerns as we go along.

Complaints that serve as clues that a UI/UX Problem exists



Section (optional)

Complaints that serve as clues that a UI/UX Problem exists

Picture (optional)

Employees refuse to do something

“Takes too long, costs too much money”

Apathetic staff

Current tools are the problem

“we just need a tool”

It’s one person’s fault

Nobody uses the tool

The people you speak to look stressed or irritated

Software Engineering Institute | Carnegie Mellon University 9

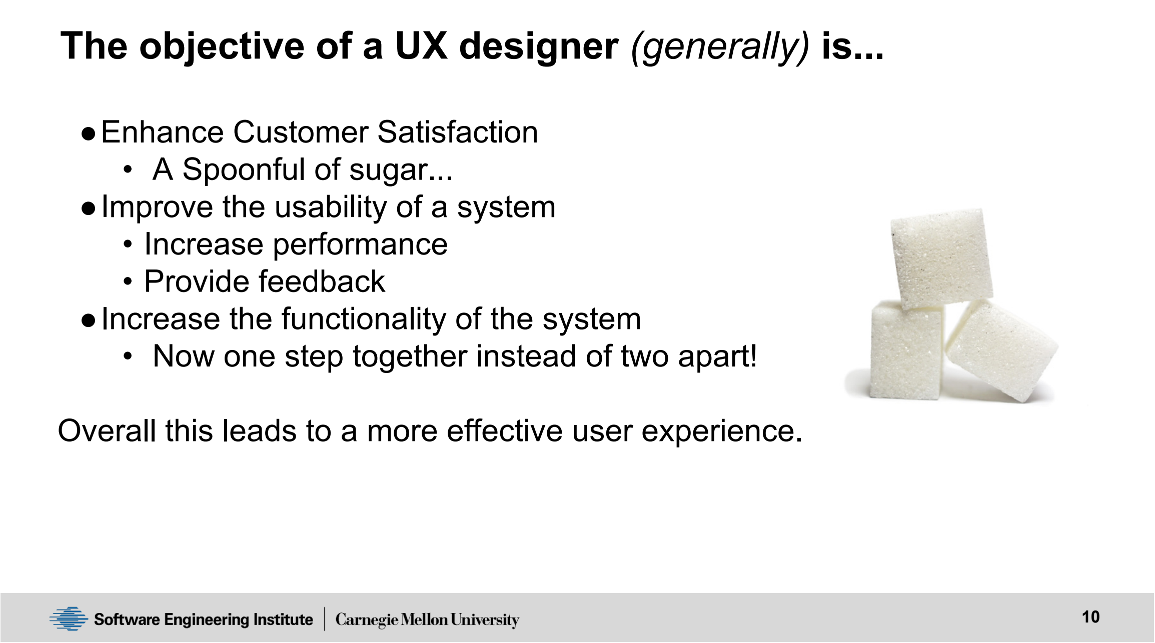
**009 So, this is a slide. It's one of my favorite slides to sort of tip off the discussion. And this is-- for those people who aren't really well versed in the details of UI/UX, they often might meet a customer who wants a web app or a piece of software developed for a particular problem. And you're listening to the customer talk. And the customer throws out stuff like these sentences. It takes too long. It costs too much money. Or, we just need a tool to slap at the problem. And you sit there, and you kind of chuckle underneath your breath. But these are some cues that you have a bigger problem than just what's at the interface level, especially when we talk about an apathetic staff.

Presenter: Yeah, and I think empathy for the end user is one of the things that we're going to focus on very heavily today. You have to keep in mind that very often when interacting on a system, the user is restricted to the software that is available to them. What we're trying to do is to understand that oftentimes the process can be contorted by the breakdowns within the tools that are available. Or, it's just a difficult tool to use, but it's the tool that there is.

Presenter: That's right.

Presenter: Right, and so being able to identify when there's breakdowns within the system, and being able to identify when the software's not serving the needs of the user in the correct way is part of being a designer and part of working within this environment.

The objective of a UX designer (generally) is...



The objective of a UX designer ***(generally)*** **is...**

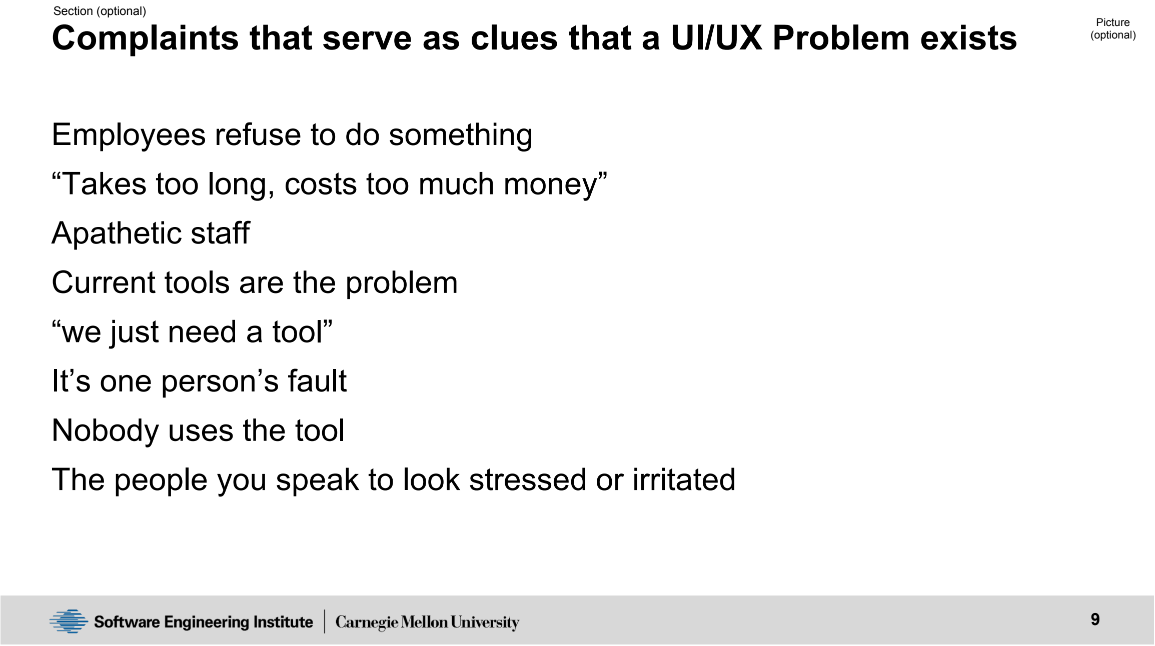
- Enhance Customer Satisfaction
  - A Spoonful of sugar...
- Improve the usability of a system
  - Increase performance
  - Provide feedback
- Increase the functionality of the system
  - Now one step together instead of two apart!

Overall this leads to a more effective user experience.

Software Engineering Institute | Carnegie Mellon University 10

**010 Presenter: No, absolutely. And I also wanted to just capitalize on that last slide there.

Complaints that serve as clues that a UI/UX Problem exists



Section (optional)

Complaints that serve as clues that a UI/UX Problem exists

Picture (optional)

Employees refuse to do something

“Takes too long, costs too much money”

Apathetic staff

Current tools are the problem

“we just need a tool”

It’s one person’s fault

Nobody uses the tool

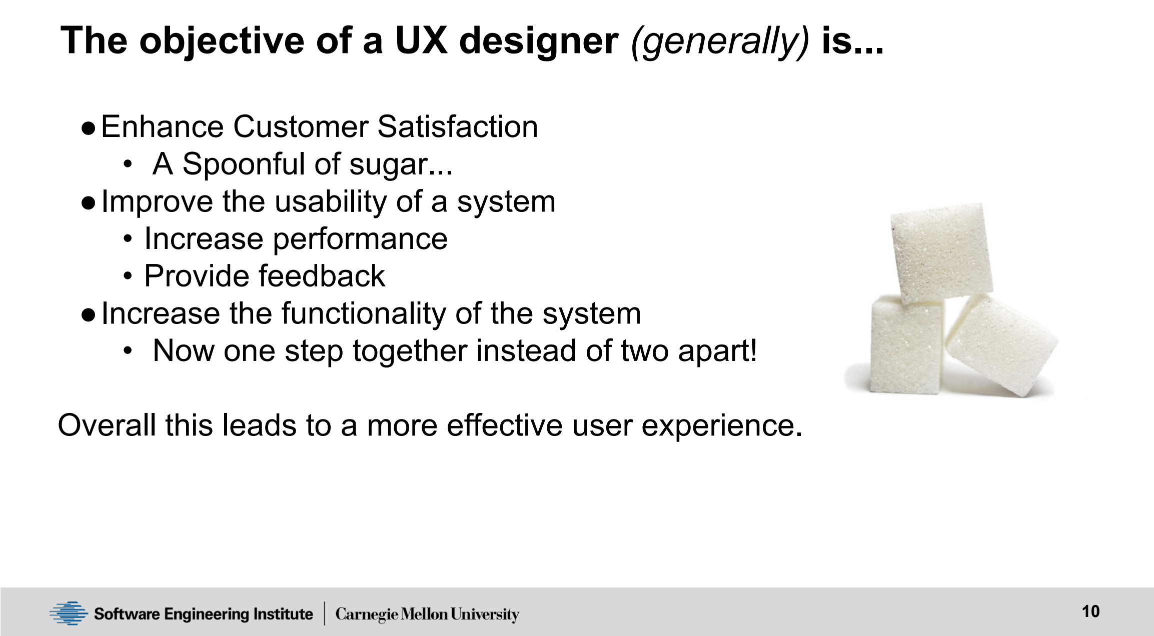
The people you speak to look stressed or irritated

Software Engineering Institute | Carnegie Mellon University

9

**009 Oftentimes, I'll come into a work environment, and I'll try to interview somebody. And they're clearly not happy with me being there interviewing them. And they're wanting to get out. And you kind of prod them about what's going on. And the emotional state that they have often is a cue that there is something much deeper than that. And you'd be surprised. A lot of times, it's not always at the UI level. It's at the UI level. The management of the team is not conducive to high productivity. And those people are very driven to produce quickly and accurately. And so, there are some encumbrances along the way. So, the problem is very multifaceted. So, these are some cues that you look for when you walk in the door. And you go oh, problem. I smell it.

The objective of a UX designer (generally) is...



The objective of a UX designer *(generally)* is...

- Enhance Customer Satisfaction
  - A Spoonful of sugar...
- Improve the usability of a system
  - Increase performance
  - Provide feedback
- Increase the functionality of the system
  - Now one step together instead of two apart!

Overall this leads to a more effective user experience.

Software Engineering Institute | Carnegie Mellon University 10

**010 Go ahead.

Presenter: So, to continue that, what is the objective of a UX designer? And once again, we're speaking in generalities. We want to focus on enhancing overall customer satisfaction. Like I mentioned earlier, oftentimes the users are required to use the system. So, we kind of go with the idea of a spoonful of sugar can make the medicine go down in some of these situations. We also want to be able to improve the usability of a system. Now, this is not just the visual aspect of it but actually evaluating the performance of the system.

Presenter: Yeah, the entire system is really key because oftentimes, one person might have stellar performance on one aspect of the tool. And then they push if over the fence to the next person who has to do something with that deliverable. And it just stops there. And so, the whole team is--

Presenter: Right, and throughout the process, you also want to be able to provide feedback. If there are breakdowns within the system, if there is some type of an error based on the way you have moved through the system, we want to be able to indicate to the user how they got to that point and how they recover from it.

Additionally, we want to increase the functionality of the system. We want to make sure that if the system supports hiding a function in between something like a button, and so we can combine multiple functions on to a single page or combine them in the back end, make sure to kind of put those two things together for the user so that they can-- if it's a step that they don't necessarily have to interact with, we don't force them to do it arbitrarily.

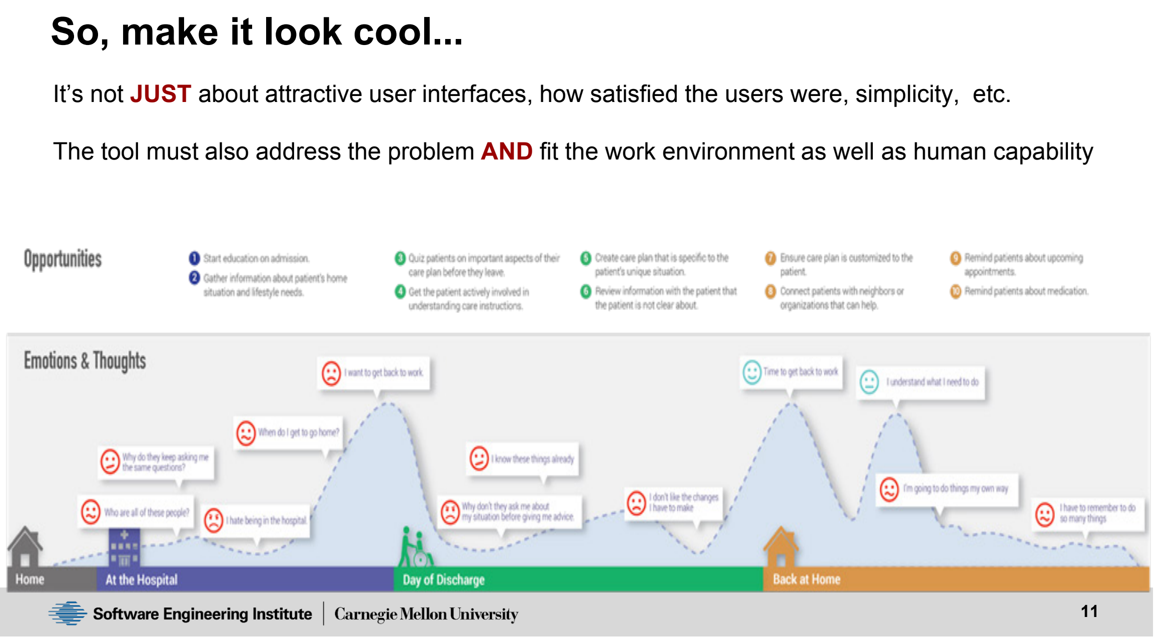
Presenter: Sure. Sure.

Presenter: But, I mean, overall--

Presenter: Yeah. Yeah.

Presenter: It leads to a better user experience.

So, make it look cool...



So, make it look cool...

It's not **JUST** about attractive user interfaces, how satisfied the users were, simplicity, etc.

The tool must also address the problem **AND** fit the work environment as well as human capability

Opportunities

1. Start education on admission.
2. Gather information about patient's home situation and lifestyle needs.
3. Quiz patients on important aspects of their care plan before they leave.
4. Get the patient actively involved in understanding care instructions.
5. Create care plan that is specific to the patient's unique situation.
6. Review information with the patient that the patient is not clear about.
7. Ensure care plan is customized to the patient.
8. Connect patients with neighbors or organizations that can help.
9. Remind patients about upcoming appointments.
10. Remind patients about medication.

Emotions & Thoughts
Who are all of these people?
Why do they keep asking me the same questions?
I hate being in the hospital.
When do I get to go home?
I want to get back to work.
Why don't they ask me about my situation before giving me advice.
I know these things already
I don't like the changes I have to make
Time to get back to work
I understand what I need to do
I'm going to do things my own way
I have to remember to do so many things
Home
At the Hospital
Day of Discharge
Back at Home

Software Engineering Institute | Carnegie Mellon University

11

**011 Presenter: Yes, and so getting back to this point. If we don't stress anything else today, this is probably one of my favorite points is it's really not about a pretty interface. And I can't tell you how many times I've looked at people's work, and it's a beautiful interface. And it's flashy. And it's got lots of style to it. And it's contemporary looking. But the end user just can't use it. And we've seen that because they didn't take enough time to evaluate ethnographically, meaning just observing the user in their natural environment, what's their competency level with respect to digital technology, what kinds of devices are they using to access that particular software, how does that really stop or halt their workflow if they have to pick up the mobile phone and use it. And that's always a key here. So, knowing that it's much broader than just the UI is really critical.

Presenter: Oh, yeah absolutely. It's not-- like it says, like we mentioned, it's not just about attractive user interfaces. The tool must address the problem and the work environment. There's been many occasions I've experienced both here at the Software Engineering Institute, and elsewhere within industry where we have the tool. We have the functions to complete a task. But the needs of the individual interacting with the system have not been properly assessed. And it doesn't matter how pretty and shiny the interface that you stick on top of it is. Once you put it in front of the user, it ends up breaking down, or they refuse to use it because it doesn't actually fit in with the process that they're participating in.

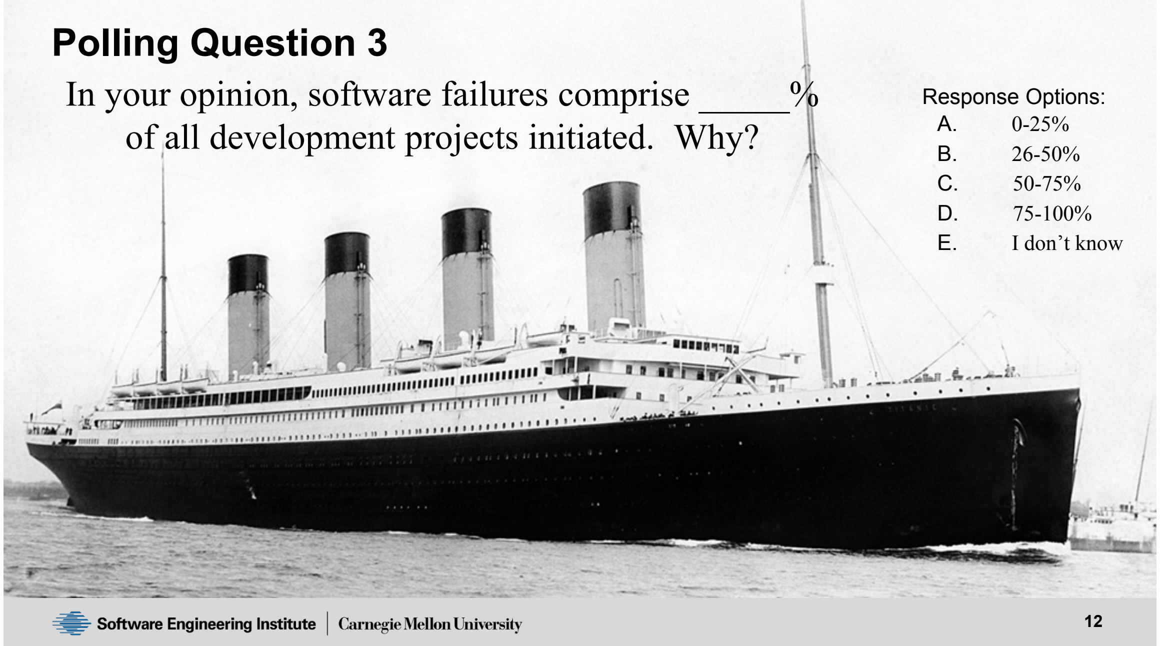
Presenter: That's right. And there's one point that I wanted to kind of go back to because I think that we didn't really spend a lot of time on it. And that is the idea of a diverse UI/UX team. Diversity is key in the sense that you will have some experts from the psychological sciences, so industrial organizational sciences, who may understand work tasking. And that's their strength, the nice part that they bring to the picture. And then they can tell you whether or not that interface actually fits with the work tasking.

And then you have designers who are amazing at styling it up or decreasing the style. So, you just get multiple perspectives. And it may seem costly at first. But in the long run, it really does save on some costs.

Presenter: Yeah, I think that having a very diverse team is something. Especially, here at Carnegie Mellon within our Human Computer Interaction Institute, they really focus within the master's program and some of the other-- the PhD program, being very diverse. People come in with very, very diverse background. And I've notice with teams that collect people from all these different areas, developers that want to get more comfortable within the design environment, psychologists or psychology majors that want to expand their skill set into development and design, and people that come from backgrounds that are unrelated to any of the three primary intersections of user experience, really they bring their past experience, their past intake into UX. And it helps to solve a lot of the problems and create really unique, interesting solutions.

Presenter: Sure. Sure.

Polling Question 3



Polling Question 3

In your opinion, software failures comprise _____% of all development projects initiated. Why?

Response Options:
A. 0-25%
B. 26-50%
C. 50-75%
D. 75-100%
E. I don't know

Software Engineering Institute | Carnegie Mellon University

12

**012 Presenter: So, that's going to lead us to our third polling question. And that is, "In your opinion, software failures comprise blank percent of all development projects initiated." And then for the why, if you have a why, we ask that you just type that into the chat box. I'm going to pass it back to you guys while we give them about ten or fifteen seconds to vote.

Presenter: Well, let's see what we got here.

Presenter: And I think that software failures is again that tricky place that maybe you're confused about. And, for example, there was a study that was conducted in 2001 or 2005, I can't remember the year, it's not very clear. But it's by Anderson. And he went out. And he did some basic web analytics at the time and discerned that after a few clicks, if the person couldn't find what they wanted in a web application for shopping, they just backed out and never came back. And to me, that's a failure because you lost your business. That seems very cut and-- black and white. But that's one example of how you can have failures.

Another failure is when you walk into an organization and you have six tools that do basically the same thing. And only one gets used. And the other five are just collecting dust in the closet. So, that was a design issue as well.

Presenter: All right, just a quick recap of the results here. We've got five percent at zero to twenty-five percent, thirty-three percent, twenty-six to fifty percent, thirty-eight percent at fifty to seventy-five percent, ten percent claiming seventy-five to a hundred percent. And we've got fourteen percent not sure. And just a couple-- if you guys want to go over a couple of the comments in the chat, if you want to read them off as well.

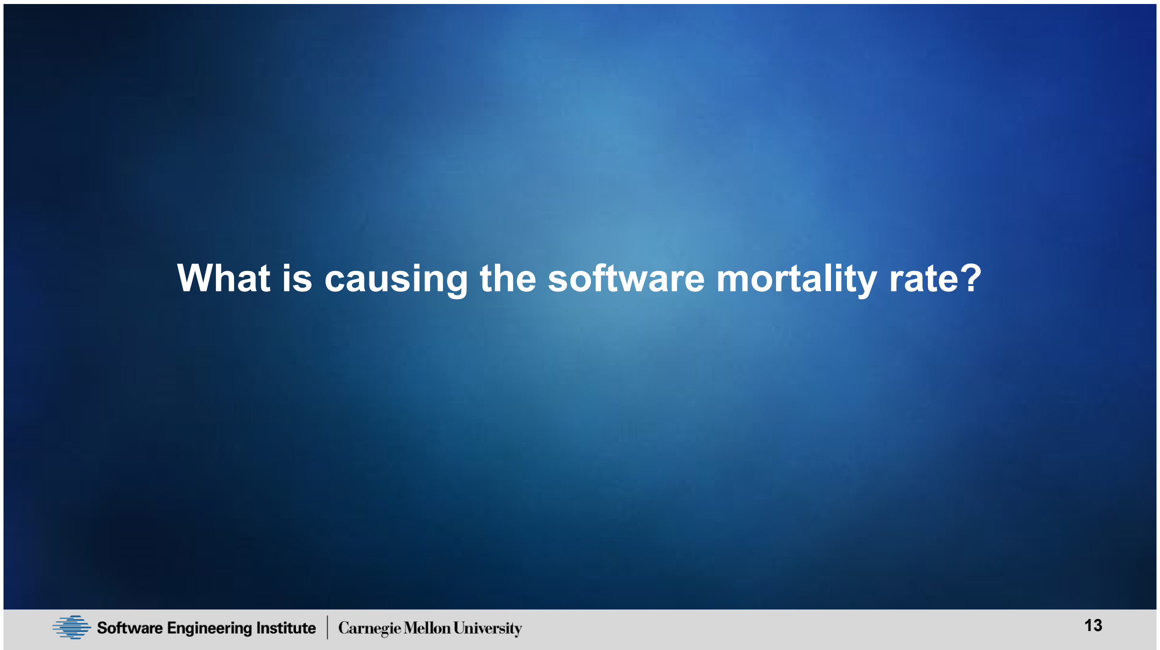
Presenter: Yeah, I have a comment here that the reason why was there were vague requirements. And software wasn't meeting those vague requirements. And I'm so glad that guest three five seven one had mentioned that because that's the crux of what we're going to talk about today. And so, another one was our technology for developing and sustaining quality software is inadequate. And that could very well be the case. We talked about some cases where we just needed to try new tools that were canned, or try new approaches, or modify certain tools is a great response.

Presenter: And I think that especially within a lot of the government circles that we find ourselves in, the concept of defining vague requirements and those vague requirements leading into a not totally complete solution is definitely something that comes up on a frequent basis. Being able to clearly define and understand what the users are experiencing, and in some cases, even getting access to the users in more secure environments can be a nightmare. It's very difficult.

Presenter: It's very hard to get there.

Presenter: Yeah, and not to mention, many of these users are mission critical type of users who are constantly on the job and doing what they're there to do. So, pulling them out and pulling them aside and interrupting them through their day can sometimes be very difficult. It's kind of a tedious experience for them. So, as a designer, we try to be understanding of that, be cognizant of their needs and the pressure that's being put on them as much as possible. And the angle is to help them do their job in a more efficient and a better manner overall.

What is causing the software mortality rate?



What is causing the software mortality rate?

Software Engineering Institute | Carnegie Mellon University 13

**013 Yeah, and as an aside, I wanted to just take a step back. And there was something else I was going to mention now that you've brought it to my memory. And that is that I had a gentleman come up to me and ask me, "Why does it matter that you're a UI/UX designer? This is cyber security. Or, this is software engineering. Who really cares about that here? Why should I care about that?" And I was thinking oh my gosh, if you only knew that it's so easy for a user to blinded because they have perceptual limits in terms of what they can see and what they can't see that if you don't know where they're failing to see something in the interface, and you have that person monitoring day in and day out on whether or not an adversary is on their network, that's a big problem. And so, the problem will guide the solutions that we offer or at least document that the problem has a numeric quantity of number of failures to detect something. So, we're actually able to find holes in the interface or other errors that they're making that actually might be a problem for cyber security.

Presenter: Yeah, and I think that's what it-- when it gets back to the original question of what's causing the software mortality rate, it's an unclear understanding or definition of the environment, the situation, the user's perspective, the way they're looking at the software in general.

The When

The When

Where does UX fit into my product cycle?

Software Engineering Institute | Carnegie Mellon University

© 2016 Carnegie Mellon University
[Insert Distribution Statement Here]

**014 Presenter: That's right. That's right. And also to capitalize on our guest who mentioned the requirements are vague, this is the lynchpin, in our opinion for how we end up with a mortality rate that's so high. When you hurry up, and you come up with the problem really fast, and you quickly throw out some requirements that may or may not address the problem, and then suddenly the aperture of possible tools that could fix the problem becomes really wide. So, starting with something very generic is not a good idea. So, if you take the time to detail and document, even though your customer who doesn't always like a lot of detail, take the time to document the problem and all of the web of sub problems underneath, that actually will guide your requirements. Your requirements will be at the level of detail such that you can select and do tradeoffs on solutions. If you don't have that problem statement very well defined, and well understood, and well vetted, it almost crashes the entire project because now you're just throwing anything you can to try to hurry it up.

And often customers want to hurry it up. But you kind of have to take a step back and explain in terms of dollars, and time, and how long it's going to take them to bug test this thing, and how it's never going to address the problem. You have to stop them from wanting to hurry it up and just say, "This is costly for you. Are you sure you want to do this?"

We have some great comments here. We had guest nine oh six four who said the webcast wasn't working for them. I hope that we can actually get that working.

Presenter: Yeah, they just may want to log in and log back out.

Presenter: Yeah, that might be a good way to do that is to log in and-- also, this guest says, "Please comment on the merits of using Agile type methods to trickle out functionality and draw out true requirements in production before the product has been completed." So, talking about iterative development and sort of how you can take a feature function set and refine it quite a bit with probably UI testing at the end of a sprint. I'm assuming at the end of a sprint for Agile.

Presenter: Yeah, I think that Agile is actually-- it's a really great environment for user experience. They're-- with a few caveats. A, you want to have the baseline established very clearly so development doesn't find themselves going down a path that we kind of veer off of, so kind of isolating the more in depth functionality from the superficial interface level considerations.

Also, I think that a lot of times, you definitely want to have a very mature development and design group, groups that are used to working with each other. Oftentimes, I've found that people try to jump right into Agile with a brand new group. And it can be very stressful and can lead to a lot of iteration. But I do think that iterative design and testing in an Agile method is very useful as you can put one particular feature or function out at a time, test to see how it's fitting into the overall environment, and then make changes as necessary instead of getting too far ahead. But in many cases, you do need an existing work product in place--

Presenter: Yeah, to help with that.

Presenter: To build off of as opposed to very many times when you do get the budget and the time to do user experience research, it's oftentimes for a product that is brand new.

Presenter: Brand new in concept, yeah.

Presenter: Yeah, so--

Presenter: And that-- also we're going to show you a little bit about a method that we use for developing the problem and then picking requirements that sort of-- that are complimentary to that particular sub problem within the problem network, and then doing the cost-benefit ratios, or the tradeoffs between using one solution versus another.

And getting back to requirements, requirements are one thing. But requirements should always generate the metrics with which you use to measure how well that interface actually addressed the problem. So, for example, if your big problem is people are just slow at doing their work in this tool, then your metric better be about speed and accuracy of the work at the end of the redesign. And so, you choose the appropriate metric there. So, the requirements drive. But it starts with the problem definition. And requirements just drive the whole rest of it.

Presenter: Right, absolutely. The quantitative aspect of user experience is one of its more--

Presenter: That's right.

Presenter: Important tools in its tool case.

Presenter: The metric aspect, it is so heavy metric if it's a good UI/UX team.

Presenter: Oh, and one thing I did want to mention that just came to me a little bit before, back to the question about Agile. Typically, in projects where I've led a team of multiple designers working in an Agile environment, I try to keep the UX team one to two sprints ahead of the development team at all times so that we-- by the time they get to the place that we are, we've basically kind of jumped on to the next thing. We can kind of circle back. But we want to have everything lined up for them and ready for them as soon as possible because kind of following within the sprint and the way certain tasks line up, sometimes that can be difficult. So, you definitely want to get a running head start and get ahead of the development group before you try to work in an Agile--

Presenter: And you'll filter out your features and functions as you go through a test. Oftentimes people say, "I don't even need that. I don't need that feature or that functionality here." It's crazy. So, then you just drop it. That's how you filter out some of the features. Not all of them, but--

Presenter: Right, which actually leads us kind of into our next topic, which is the when.

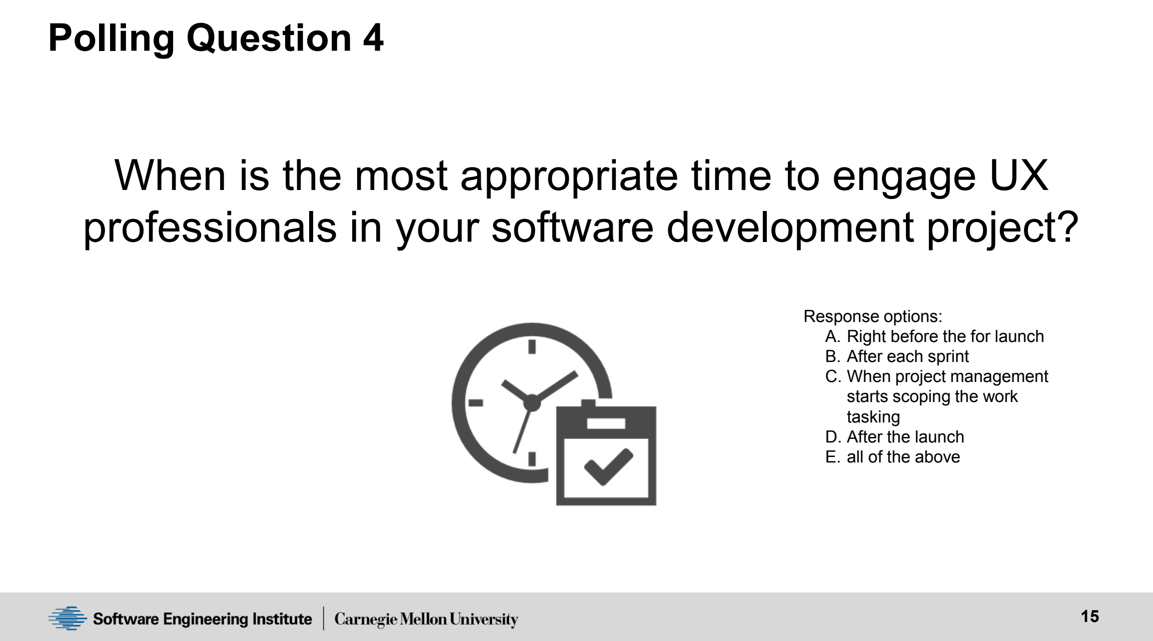
Presenter: The when, my favorite.

Presenter: Yeah, so where does--

Presenter: Whoa, the time.

Presenter: Yes, within the whole schedule, because we have all the time, and money, and effort in the world just to focus on this one thing, where does UX actually fit into my product cycle.

Polling Question 4



Polling Question 4

When is the most appropriate time to engage UX professionals in your software development project?

Response options:

A. Right before the for launch
B. After each sprint
C. When project management starts scoping the work tasking
D. After the launch
E. all of the above

Software Engineering Institute | Carnegie Mellon University 15

**015 Presenter: And that's going to lead us to our next polling question, which is, "When is the most appropriate time to engage UX professionals in your software development project?" We'll give you about fifteen seconds to vote. While you're voting, let's get an audience question from Joseph wanting to know, "I'm starting a new UX team at my organization. And I'm putting together a list of potential tools we could use. How do I know if I need eye tracker for my user testing, eye tracker?"

Presenter: So, that's-- yeah, that's an interesting question. I mean it would definitely be defined by whether or not the tools that you're creating justified that type of process.

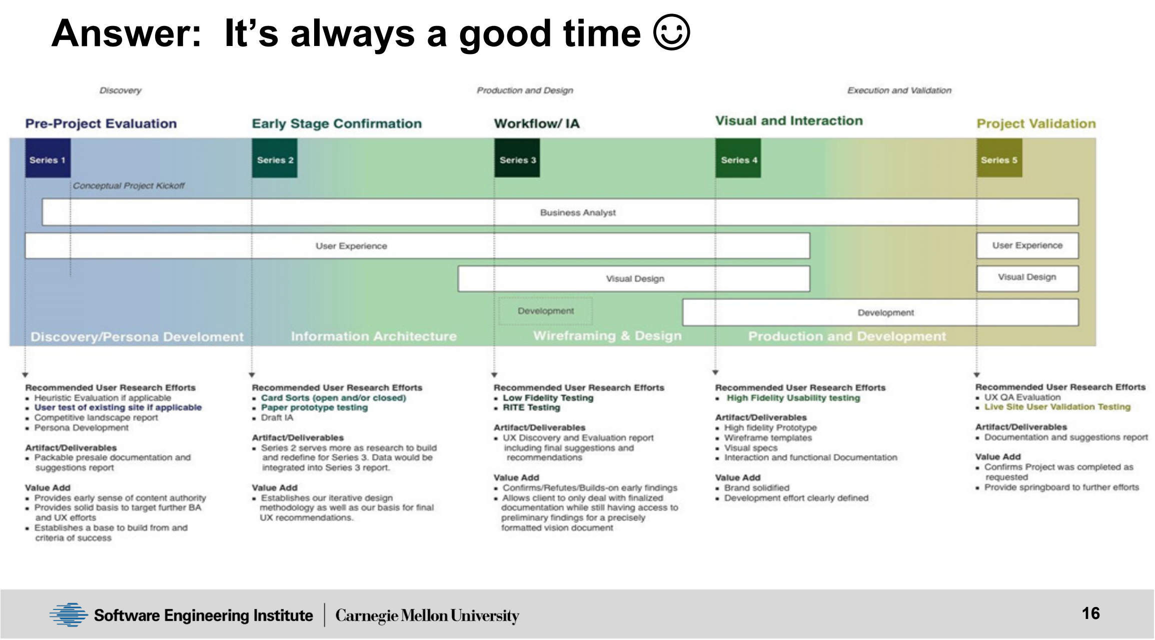
Presenter: That's right. It's a metric. You go back to the metrics. So, if the problem is that maybe the person isn't seeing something, and that's causing a lot of errors in the interface, then sometimes use it as secondary measure to say well why aren't they clicking on this button. Well, they're not even seeing it. So, use an eye tracker to sort of track their visual state in moment by moment time.

Presenter: Right, and many times, organizations kind of fall back into wanting to set up all their software and hardware. Depending on how frequently you need that hardware, and your access to professionals who have been guided in the correct usage of the software, there are also many facilities that can provide you with your own testing facilities. They have access to the software. They have technicians that are licensed to use the software, not licensed but taught to use the software in an accurate method. So, it really would depend on your needs and your availability to use the equipment. But I've found that a lot of times, if you have that extra software, and you have that extra hardware, and you can kind of play with it on the side when you're not doing testing, it can give you some ideas of how to expand your functionality outside of your immediate testing needs.

Presenter: That's right.

Presenter: Great, so we'll go back to our results. We got eight percent after each development cycle. And the question again was, "When is the most appropriate time to engage UX professionals?" Ten after the development cycle, fifty percent when the project management starts scoping the work tasking. And then we had forty-two percent at all of the above.

Answer: It’s always a good time ☺



Answer: It’s always a good time ☺

Recommended User Research Efforts
- Heuristic Evaluation if applicable
- **User test of existing site if applicable**
- Competitive landscape report
- Persona Development

Artifact/Deliverables
- Packable presale documentation and suggestions report

Value Add
- Provides early sense of content authority
- Provides solid basis to target further BA and UX efforts
- Establishes a base to build from and criteria of success

Recommended User Research Efforts
- **Card Sorts (open and/or closed)**
- **Paper prototype testing**
- Draft IA

Artifact/Deliverables
- Series 2 serves more as research to build and redefine for Series 3. Data would be integrated into Series 3 report.

Value Add
- Establishes our iterative design methodology as well as our basis for final UX recommendations.

Recommended User Research Efforts
- **Low Fidelity Testing**
- **RITE Testing**

Artifact/Deliverables
- UX Discovery and Evaluation report including final suggestions and recommendations

Value Add
- Confirms/Refutes/Builds-on early findings
- Allows client to only deal with finalized documentation while still having access to preliminary findings for a precisely formatted vision document

Recommended User Research Efforts
- **High Fidelity Usability testing**

Artifact/Deliverables
- High fidelity Prototype
- Wireframe templates
- Visual specs
- Interaction and functional Documentation

Value Add
- Brand solidified
- Development effort clearly defined

Recommended User Research Efforts
- UX QA Evaluation
- **Live Site User Validation Testing**

Artifact/Deliverables
- Documentation and suggestions report

Value Add
- Confirms Project was completed as requested
- Provide springboard to further efforts

Software Engineering Institute | Carnegie Mellon University 16

**016 Presenter: Well, forty-two percent, you're right. This one actually was not an opinion question.

Presenter: No, we were just playing with you.

Presenter: It's always a good time. One of the things that we have mapped out on this particular slide here is the idea that UX can be fit into multiple different stages, even if you are midstream already. To be able to get in there, evaluate what the current situation is of the system, being able to prevent what could possibly be a misstep further down the line, that's always something that's available within the process, and even if you have existing tools. You have heuristic evaluations, where you can go through the application with a fresh set of eyes, a few different designers looking at it, everybody making their comments, consolidating their concepts back together and putting together a report, documenting the requirements, getting to know the users.

And you'll have to pardon the illustration. It is designed from a UX perspective. Now, I realize that the development box there should be oftentimes much, much, much longer. But PowerPoint is only so wide. So, we don't want to go too crazy with that.

Presenter: And to capitalize on what you were saying, we have actual research evidence that from various publications in the early 2000s and late 1990s and actually up in mid-2000s that the earlier you start working with the UI/UX, the more cost effective the end result will be. In other words, we will disentangle the problems early. And that way you're not compounding those problems with code if they're not fixed early. So, it just becomes a bigger and bigger ball of wax.

And the other failure that I think a lot of companies forget about is at post launch. After it's been released, often a user's requirements will shift because technology is moving pretty quickly. And you know that-- how long did the Microsoft products sometimes just sit there and not be updated. And you were getting all these whiz-bang tools, and other tools that could do just as well. And they decided to change and update their features. So, if you want to stay competitive in the market, you need to do post launch requirements assessment and see how well that tool is actually working.

Presenter: Absolutely, which can lead us into our next topic.

Artifacts of the Process

Artifacts of the Process

Inform and build empathy

Software Engineering Institute | Carnegie Mellon University

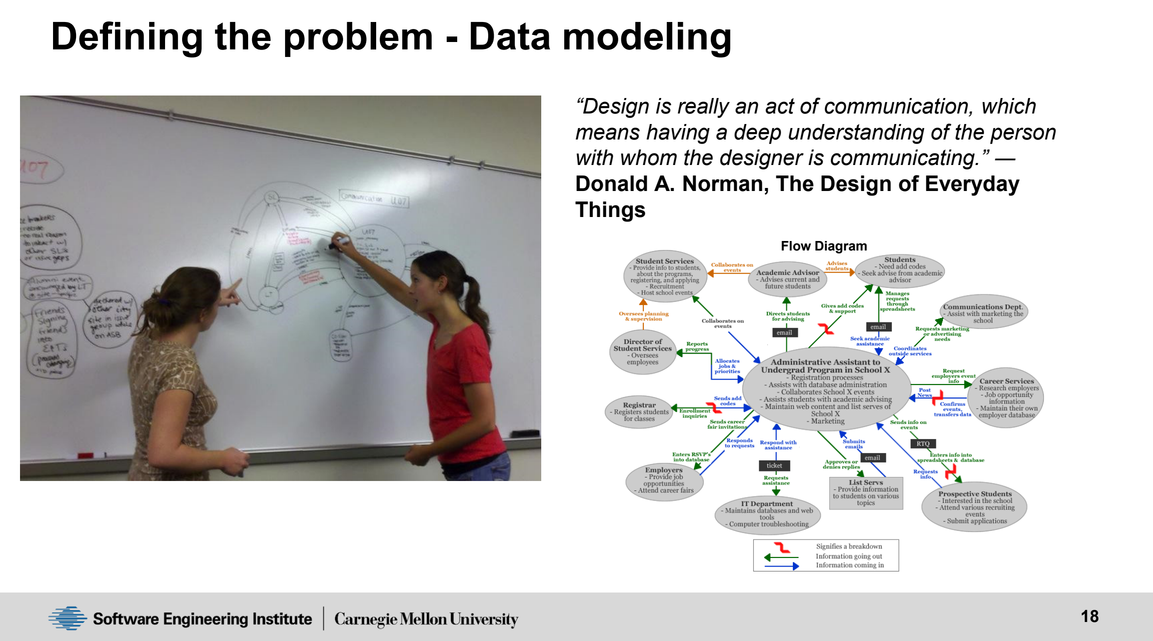
© 2016 Carnegie Mellon University
[Insert Distribution Statement Here]

**017 Presenter: Oh yeah.

Presenter: Which is the artifacts of the process. So, these are the artifacts that are created through the initial discovery phases, can be referenced at later stages, post development, to see well, this is the direction we went in, why did we choose that. But the overall goal of most of the artifacts that are generated through the user experience process are designed to obviously inform the designs and interactions but also build empathy for the end users.

Presenter: Right, that's right. So, why is empathy really the crux of UI/UX?

Defining the problem - Dat a modeling



Defining the problem - Data modeling

"Design is really an act of communication, which means having a deep understanding of the person with whom the designer is communicating." —
Donald A. Norman, The Design of Everyday Things

Flow Diagram
Student Services
- Provide info to students, about the programs, registering, and applying
- Recruitment
- Host school events
Collaborates on events
Academic Advisor
- Advises current and future students
Advises students
Students
- Need add codes
- Seek advise from academic advisor
Oversees planning & supervision
Collaborates on events
Directs students for advising
email
Gives add codes & support
Manages requests through spreadsheets
email
Communications Dept.
- Assist with marketing the school
Seek academic assistance
Requests marketing or advertising needs
Coordinates outside services
Director of Student Services
- Oversees employees
Reports progress
Allocates jobs & priorities
Administrative Assistant to Undergrad Program in School X
- Registration processes
- Assists with database administration
- Collaborates School X events
- Assists students with academic advising
- Maintain web content and list serves of School X
- Marketing
Request employers event info
Career Services
- Research employers
- Job opportunity information
- Maintain their own employer database
Post News
Confirms events, transfers data
Registrar
- Registers students for classes
Sends add codes
Enrollment inquiries
Sends career fair invitations
Sends info on events
Enters RSVP's into database
Responds to requests
Respond with assistance
Submits emails
RTQ
Enters info into spreadsheets & database
ticket
Approves or denies replies
email
Requests info
Employers
- Provide job opportunities
- Attend career fairs
Requests assistance
List Servs
- Provide information to students on various topics
IT Department
- Maintains databases and web tools
- Computer troubleshooting
Prospective Students
- Interested in the school
- Attend various recruiting events
- Submit applications
Signifies a breakdown
Information going out
Information coming in

Software Engineering Institute | Carnegie Mellon University 18

**018 It sounds kind of hokey to people who are very black and white, very scientific and rigorous. It seems too soft for them. But ultimately, there's a really good fundamental reason for the empathy.

Presenter: Yeah, really understanding, putting yourself in the perspective of the user, understanding the environment that they find themselves in, the tasks that they have to complete, it allows you to A, step out of yourself and your own personal biases, in many cases and look at the problem in a more holistic manner that allows you to understand why something that might-- one of our mantras at the HCII here is I am not the user. I can look at a lot of applications, and to me it looks super simple. But if I put that same application in front of my grandmother, she might be really super confused by it. And so, when I'm designing applications that somebody like my grandmother might interact with, I need to be understanding that my advanced understanding of how to interact with the system doesn't necessarily extend to the rest of the world.

Presenter: That's right. That's right.

Presenter: So, how do we get to that point?

Presenter: Oh, gosh. And here's where I think a good UI/UX team can really shine. You have constraints as a person, a customer who needs to purchase software. You need to have the software done by a certain date. And we will-- the UI/UX team will work with project management heavily to try to come up with timelines. Well, we need all this data to understand the problem, and get good requirements, and figure out the testing schedule, and figure out where we fit in the sprint cycles. And if we can't-- my favorite is when you see in the field that the UI/UX team had two weeks to do something. And they picked a method, an approach, to getting information that usually takes four to six weeks.

Presenter: Right.

Presenter: It's like, why are we doing this? So, you can read the quality of the report is very low. And it never really talks about the problem. It's because of doing things in haste. And so, you can actually creatively come up with new strategies for that. So, if you can't ever get to the actual end user, there are other strategies that we use, such as we go in to forum spaces of competitive products that have very similar UI/UXs. And we read about the features that we are interested in designing to fit the problem. And we take all that data, and diagram it, and try to make sense of it if nothing else is there.

Presenter: Right, and that's the first step. You want to reach out. And you want to define the problem.

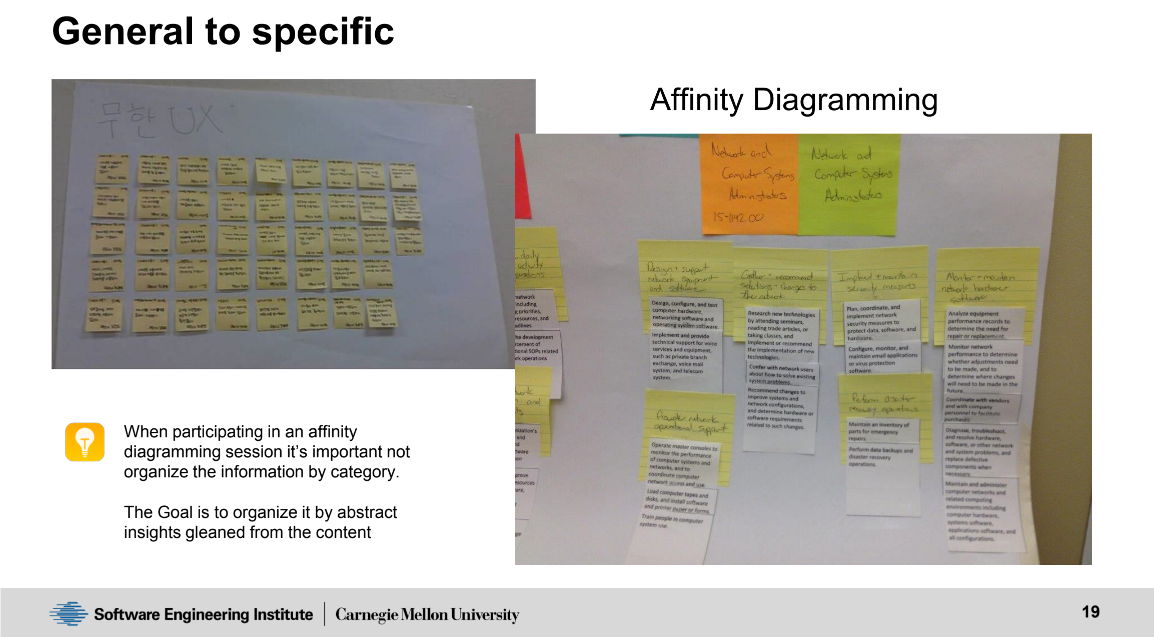
Presenter: Define the problem is right.

Presenter: Don Norman, the design of everyday things, design is really an act of communication, which means having a deep understanding of the person with whom the designer is communicating. And what you're seeing in the photograph on the screen right there is two designers working on a flow diagram, which is a diagram that talks about all the different systems, and people, and actors, and how they connect, and the way they connect.

What you're doing is through your interviews, you're trying to access each one of the different members of the community that your system might be touching, understanding how they interact with each other, understanding how they interact with data outside of the system or different systems. You're just trying to understand, at the end of the day how are all these different people touching each other and interacting with each other. A lot of times once you have a very clear understanding of how people are interacting on a day to day basis, it can start to surface where the breakdowns are happening. If somebody doesn't understand how to communicate data from person A to person D, and it's because it needs to filter through person C, maybe you need to provide an opportunity for person A and person D to connect more accurately on their own. Or, maybe there's too much communication, which is definitely--

Presenter: Oh, that's the truth, yes. And that's another-- thank you for bringing that up. Another feature or tip that you have a good UI/UX team is that a good UX designer is actually looking through the research publications to find the latest trends on software design, or software features, or software layouts, or what have you, and the negative or adverse effects it has on the user population. And there are plenty to be had. But what we see is often in development, they don't take the time to read it. And here they are throwing social media--

General to specific


무한 UX
Network and Computer Systems Administrators 15-1142.00
Network and Computer Systems Administrators

General to specific

Affinity Diagramming

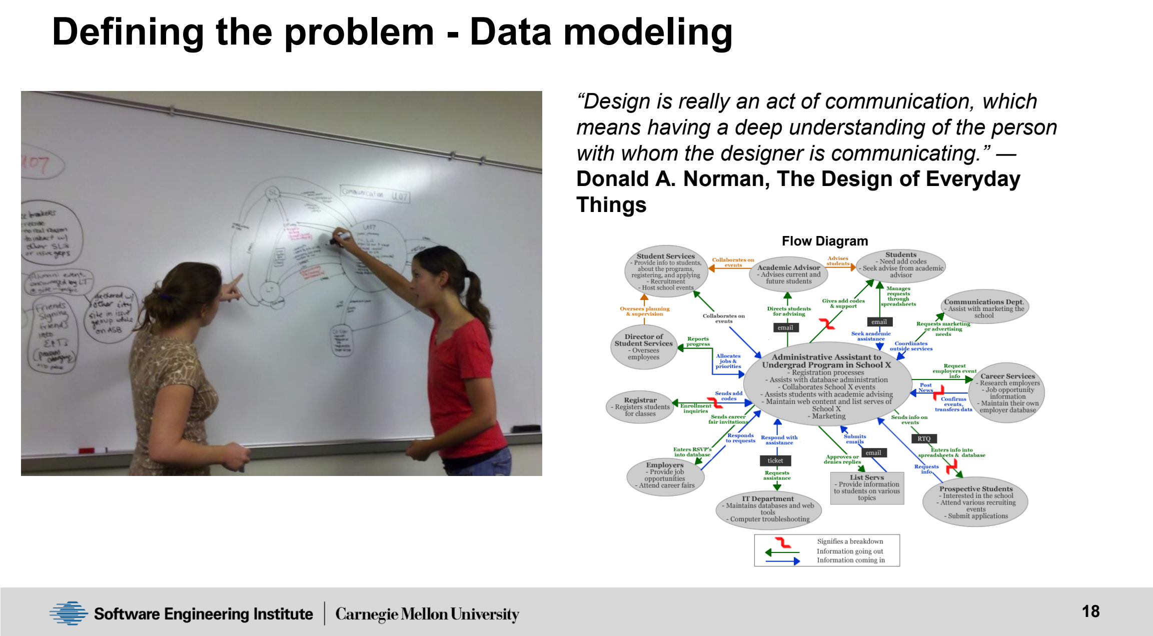
When participating in an affinity diagramming session it's important not organize the information by category.

The Goal is to organize it by abstract insights gleaned from the content

Software Engineering Institute | Carnegie Mellon University 19

**019 Into a job task that actually-- social media distracts you, disrupts you--

Defining the problem - Dat a modeling



Defining the problem - Data modeling

“Design is really an act of communication, which means having a deep understanding of the person with whom the designer is communicating.” — **Donald A. Norman, The Design of Everyday Things**

Flow Diagram

Student Services
- Provide info to students, about the programs, registering, and applying
- Recruitment
- Host school events
Collaborates on events
Academic Advisor
- Advises current and future students
Advises students
Students
- Need add codes
- Seek advise from academic advisor
Oversees planning & supervision
Collaborates on events
Directs students for advising
email
Gives add codes & support
Manages requests through spreadsheets
email
Communications Dept.
- Assist with marketing the school
Requests marketing or advertising needs
Seek academic assistance
Coordinates outside services
Director of Student Services
- Oversees employees
Reports progress
Allocates jobs & priorities
Administrative Assistant to Undergrad Program in School X
- Registration processes
- Assists with database administration
- Collaborates School X events
- Assists students with academic advising
- Maintain web content and list serves of School X
- Marketing
Request employers event info
Career Services
- Research employers
- Job opportunity information
- Maintain their own employer database
Post News
Confirms events, transfers data
Registrar
- Registers students for classes
Sends add codes
Enrollment inquiries
Sends career fair invitations
Responds to requests
Enters RSVP's into database
Respond with assistance
Sends info on events
Submits emails
RTQ
Employers
- Provide job opportunities
- Attend career fairs
ticket
Requests assistance
Approves or denies replies
email
Enters info into spreadsheets & database
Requests info
List Servs
- Provide information to students on various topics
IT Department
- Maintains databases and web tools
- Computer troubleshooting
Prospective Students
- Interested in the school
- Attend various recruiting events
- Submit applications
Signifies a breakdown
Information going out
Information coming in

Software Engineering Institute | Carnegie Mellon University 18

**018 Gets you to split your cognition. And your memory is often compromised. And yet, you want those people to drill down in the weeds for hours and hours. And you're trying to get them to be disrupted. Not good, not good for that job role. It works well for sales, but it doesn't work well for like reverse engineering. And so, adding social media components to their natural tools, not a good idea. But it was sexy, and it was exciting. It was progressive. And they thought it was a good idea. So, your UI/UX team is actually-- should be looking for the negative effects of features that are put into software tools.

Presenter: Right, very often I-- especially in situations like here at the Software Engineering Institute, we have amazingly talented developers. These men and women are absolutely amazing at what they do and how they do it. And they are creating incredibly powerful software. And when you talk about interface design, they want to have all those features and functions available to a user. They want ever user to be able to be a super user. And there are situations--

Presenter: Feature-itis is what we call that.

Presenter: Yeah, and sometimes that's just not reasonable. Sometimes, you think about the eighty-five/fifteen rule. Eighty-five percent of the users will only use X percentage of this tool's actual functionality. So, we need to make sure to be able to provide that functionality, but as a side item, as a different layer to the interface, as opposed to making that be the primary focus and front and center. So, it's trying to determine what is--

Presenter: What's reasonable.

Presenter: Yeah.

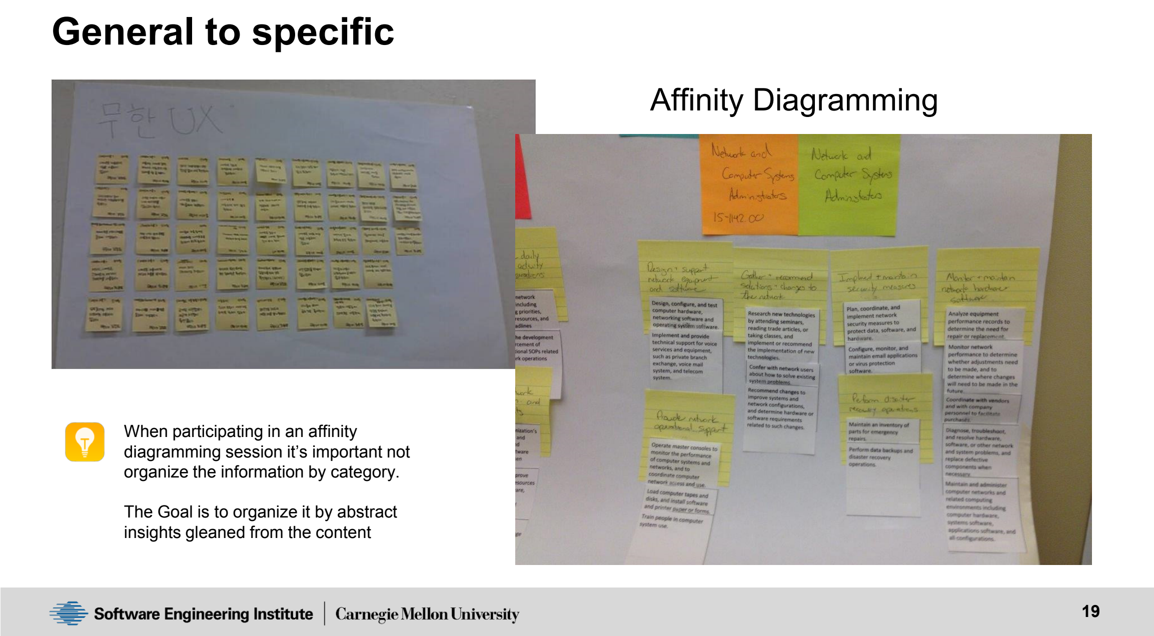
Presenter: And here's another tip for not a very good UI/UX team is when they're always interested in dumbing down the interface, making it too simple because--

Presenter: Right, there's definitely a--

Presenter: The complexity has to match. And here's a good case example. A while ago, I-- several years ago, I was designing, or helping design, a HUD display for a Boeing triple seven, 777. And we automatically, we, my research team, we were more human factors than HCI. And we said, "Well, the simpler, maybe the less clutter in the interface, maybe the more likely they are to make less mistakes with navigation." And when we ran a few really experienced pilots through it, they said, "Are you kidding? Where did that TCAS? Where did that tunnel guidance go? I need that back up. I use that to guide it in." And so, they liked it full on full clutter because every feature, they've trained over the years to use. And to an outsider, it looks completely like a smattering of stuff and symbology. To them, it was important and very useful. So, it wasn't cluttered to them at all. We thought it was extremely cluttered.

So, simplifying and interface shouldn't happen in the absence of the user who will tell you immediately, "That's not a good idea. I need that to perform my work."

General to specific



General to specific

Affinity Diagramming

When participating in an affinity diagramming session it's important not organize the information by category.

The Goal is to organize it by abstract insights gleaned from the content

Software Engineering Institute | Carnegie Mellon University 19

무한 UX
Network and Computer Systems Administrators 15-1142.00
Network and Computer Systems Administrators
Design + support network equipment and software
Gather + recommend solutions + changes to the network
Implement + maintain security measures
Monitor + maintain network hardware + software
Perform disaster recovery operations

**019 Presenter: Absolutely. Absolutely. And kind of getting a better understanding of what the user needs, one of the first things and one of the important things is sitting down, conversing with those users and getting an understanding of the world that they exist. Take notes. We oftentimes, when appropriate, we'll record our conversations with users, use that to come back and create our notes. And then we get into everybody's favorite part of user experience design, which is Post-it notes.

Presenter: That's right, so every comment that you get from an interviewee, we write it down. And we document exactly where in the written transcript it came from. And we pile them up, right? Is that true? We pile them up. And then you and I, if we had this stack of affinity-- we call them affinity notes, you and I would just throw them on a board and start arranging sort of from the bottom up. We don't have any preconceived notions. We would just go ahead and start making arrangements.

Presenter: Right, and you're trying to--

Presenter: Laugh at them.

Presenter: Yeah, absolutely, you're trying to make loose associations in the beginning. You're trying to figure out, without categorizing, which is something you definitely want to avoid, try to find different things that draw relationships, maybe things that are leading to a negative tone, or points where the user is missing information or whatever. You have to look at the comments. And you have to think on it in a slightly higher level. You try not to just say, "Well, this one's talking about a computer. And this one's talking about a computer. They belong together." Yeah, that doesn't--

Presenter: No, that's actually a bad UI idea.

Presenter: And this isn't an exercise that I would say is exclusively for user experience designers or researchers at all. I think that it's always a really interesting time to get developers, product managers--

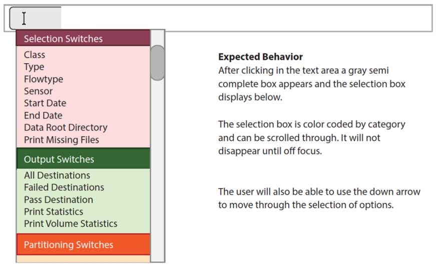
Presenter: People from their team too.

Presenter: Business analysts, get several people involved in it because what happens is you develop a shared understanding of the concepts. You get a shared understanding of where the breakdowns are happening and where solutions can emerge through this process.

Presenter: Right, and the complexity is not lost there. So, the complexity stays, which is a lovely thing. And you want that complexity so that you can actually talk about a good requirement to address that problem.

Personas, journey maps and other design documentation

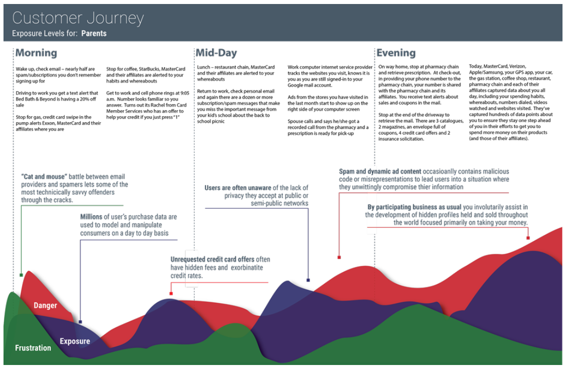
Personas, journey maps and other design documentation



Expected Behavior
After clicking in the text area a gray semi complete box appears and the selection box displays below.

The selection box is color coded by category and can be scrolled through. It will not disappear until off focus.

The user will also be able to use the down arrow to move through the selection of options.



Customer Journey
Exposure Levels for: **Parents**

Morning

Wake up, check email – nearly half are spam/subscriptions you don't remember signing up for

Driving to work you get a text alert that Bed Bath & Beyond is having a 20% off sale

Stop for gas, credit card swipe in the pump alerts Exxon, MasterCard and their affiliates where you are

Stop for coffee, StarBucks, MasterCard and their affiliates are alerted to your habits and whereabouts

Get to work and cell phone rings at 9:05 a.m. Number looks familiar so you answer. Turns out its Rachel from Card Member Services who has an offer to help your credit if you just press "1"

Mid-Day

Lunch – restaurant chain, MasterCard and their affiliates are alerted to your whereabouts

Return to work, check personal email and again there are a dozen or more subscription/spam messages that make you miss the important message from your kid's school about the back to school picnic

Work computer internet service provider tracks the websites you visit, knows it is you as you are still signed-in to your Google mail account.

Ads from the stores you have visited in the last month start to show up on the right side of your computer screen

Spouse calls and says he/she got a recorded call from the pharmacy and a prescription is ready for pick-up

Evening

On way home, stop at pharmacy chain and retrieve prescription. At check-out, in providing your phone number to the pharmacy chain, your number is shared with the pharmacy chain and its affiliates. You receive text alerts about sales and coupons in the mail.

Stop at the end of the driveway to retrieve the mail. There are 3 catalogues, 2 magazines, an envelope full of coupons, 4 credit card offers and 2 insurance solicitation.

Today, MasterCard, Verizon, Apple/Samsung, your GPS app, your car, the gas station, coffee shop, restaurant, pharmacy chain and each of their affiliates captured data about you all day, including your spending habits, whereabouts, numbers dialed, videos watched and websites visited. They've captured hundreds of data points about you to ensure they stay one step ahead of you in their efforts to get you to spend more money on their products (and those of their affiliates).

"Cat and mouse" battle between email providers and spamers lets some of the most technicically savvy offenders through the cracks.

Millions of user's purchase data are used to model and manipulate consumers on a day to day basis

Unrequested credit card offers often have hidden fees and exorbinatite credit rates.

Users are often unaware of the lack of privacy they accept at public or semi-public networks

Spam and dynamic ad content occasioanlly contains malicious code or misrepresentations to lead users into a situation where they unwittingly compromise thier information

By participating business as usual you involutarily assist in the development of hidden profiles held and sold throughout the world focused primarily on taking your money.

Software Engineering Institute | Carnegie Mellon University 20

**020 Presenter: Absolutely, absolutely. And there's more to it when building out and documenting a system. We have personas, journey maps, and other elements. I mean the persona, and this is something I see oftentimes a little bit of confusion when comparing traditional personas, like a marketing persona, versus a user experience persona. Traditionally, a persona within the realm of user experience is being used to create empathy. Once again, the E word again.

Presenter: The E word.

Presenter: When you're sitting in a room with a group of developers and project managers and anybody else involved with a particular project, you want to be able to explain why we need one particular function that might take a very long time. It might take a significant amount of time and effort. You need to be able to say we've identified these specific groups. This is where the breakdowns are happening. We're seeing this across multiple users. Think from the perspective of the person that you've documented. This is why or how we should do a specific thing. You can use that to build artifacts like an impact versus effort matrix.

Presenter: That's actually a really nice quantitative decision maker for feature functionality picking. As long as the items that you start with in your matrix, they all address the problem. All of them have to address the problem. That's sort of a subjective opinion. But at that point, once you get your collection of possibilities, then you start to do that kind of quantitative choosing one over the other, like a cost-benefit ratio.

Presenter: Absolutely. Also, artifacts like journey maps and things like that to allow you to understand how many different times throughout the day a user encounters a specific breakdown or something like that.

Presenter: And that comes directly from the affinity work.

Presenter: Right, exactly.

Presenter: So, we generate these artifacts from that first artifact. So, you're sort of compounding. So, if you have garbage in your affinity, then you'll have garbage for your other stuff. So, you have to be very careful about that.

Presenter: Absolutely. I would however say that when we get into things like personas, personal experience, I've found more success the closer I keep my personas to actual human beings that I've encountered through my research. I think it adds a little bit of validity to the overall process. This is like eighty-five percent a person that I actually talked to. This person really kind of embodied a lot of the breakdowns that we're having within the system or a lot of things that are good within the system that we need to make sure that we're maintaining through the redesign process.

Presenter: So, I'm going to-- well, actually-- We have a guest who asked a question, "Do you, at SEI and for Shane, have any downloadable tools for specifying UX requirements to engineers and any UI tool that you have created for wire framing?" So, wire framing is canned. I mean they're often canned, aren't they, because we used to use all kinds of stuff?

Presenter: Yeah, I would say at least, depending on your comfort level and what you have access to, personally, I kind of prefer Axure. I like to build click-through prototypes from that.

Presenter: GUI Design Studio was one that we used to use. PowerPoint, I've used PowerPoint.

Presenter: Some people use PowerPoint. I'm not a big PowerPoint. But yeah, Balsamic, the Adobe Create Suite, obviously, you can do click-throughs in InDesign. A lot of people use Adobe Illustrator. I mean it really just kind of depends on what software and tools you have available to you. So, we don't have anything specific that we use in that instance.

Presenter: Yeah, like nothing we developed in house.

Presenter: It really comes down to individual preference.

Presenter: We just end up using what's out here marketable. But the other part of this question was interesting, so downloadable tools for specifying UX requirements. And this is like-- I think this is the perfect combination of engineering, art, science, and research. I mean essentially, what you do is you-- when we look at our models of the problem that we get from that affinity diagram, then we start to choose requirements that tend to address-- and they sometimes come out of insights, which is an aspect of that affinity diagramming if you keep going with that approach because affinity diagram, we just showed you one step, but it keeps going forever. And so, there's a piece of that affinity diagramming process that we use to-- that can help enable us to generate good requirements. But we always cross validate our requirements back with the problem statement and all the complexity of the sub problems involved.

Remember, it's a morass and an entanglement of sub problems that often lead to the big problems you see. So, it's not just this person is slow. They're slow because they're exhausted because the tool that they're using isn't working. And their manager is pushing them to be productive. And that's actually their reward cycle internally is to hurry it up and get a lot of really good quality out there. But they can't because the policies won't allow them to-- the policies of the organization won't allow them to change tools or create a great sandbox to get their work done really quickly. So, they're exhausted.

So, you might see oh, they're exhausted. But it's really a web of problems. It involves their management. It involves their policies in the organization, the culture, themselves, their abilities, the tools.

Presenter: And--

Presenter: Sorry.

Presenter: And no, absolutely. And I would say as far as tools for requirements, kind of touching back on the question, at least here, primarily we function using the Atlassian suite, so using something like Jira to document our user requirements, and then to kind of--

Presenter: For the Agile process, right? So, Jira is like an Agile tool to sort of layout the work tasking and the requirements, correct?

Presenter: It is. It's primarily used in an Agile environment. But it would be perfectly fine in a waterfall environment as well. They're actually more the tools that we use. They're add ons to kind of extend the functionality to break it down into sprints and Kanban boards and that type of stuff. That's usually an add on. But it can function either way.

I would say one of the main focuses, or most important elements, because many times, you could just do it within a Word document-- yeah, you just want to make sure that you have clearly defined user requirements. Sit down--

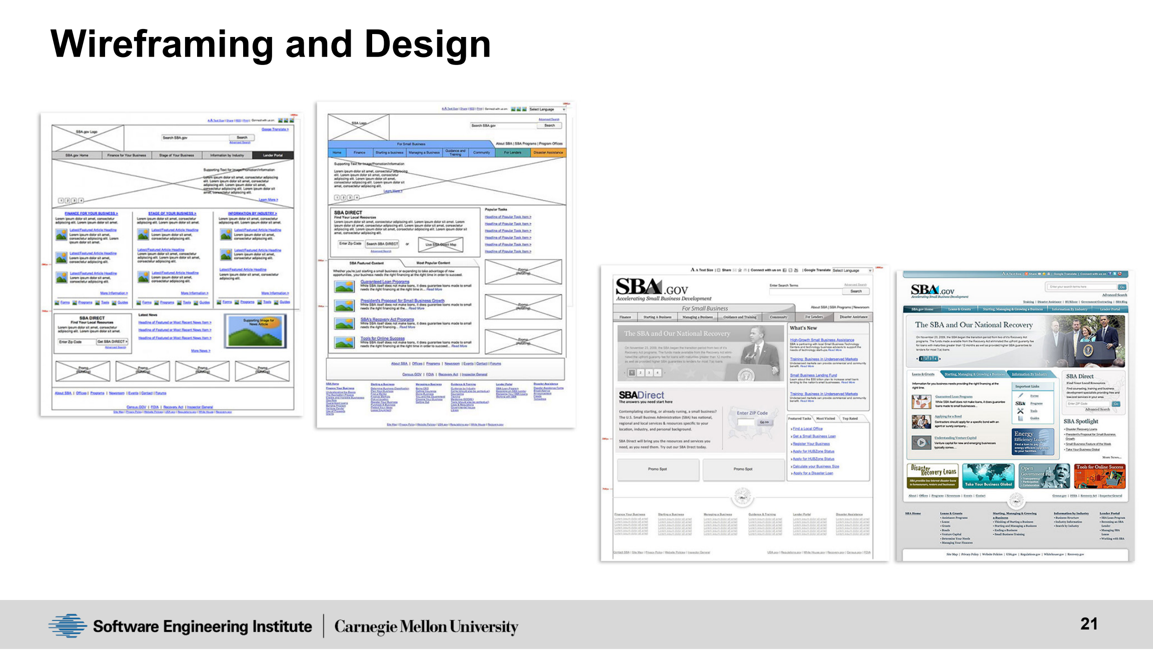
Presenter: And the reason why--

Presenter: Is because you want to make sure that you have each one of those broken down to their component elements and the needs of the user so that they're addressed somewhere because they don't always fit into the day-to-day development cycle. Sometimes, you're like, "Oh yeah, that's a side thing." You want to make sure that each one of these elements is actually noted down.

And then it's really nice to go through and document with the developers the technical requirements to understand--

Presenter: Yeah, that's the key is can the technical requirement be matched to the original requirement the user has or the system has, which is-- that's the hard part. Yeah.

Wireframing and Design



Wireframing and Design

Software Engineering Institute | Carnegie Mellon University

21

**021 Well, anyway, if we didn't

answer your question, please email us later because we spent five minutes on that. But anyway, go ahead into the wire framing.

Presenter: Sure, and then the next part that everybody wants to see, wire framing.

Presenter: Pretty things.

Presenter: Yes, everybody wants to see the wireframes. The example ones that I have up here are a little bit further down the process than you would initially start. You definitely, once again, you want to go general to specific. So, in the beginning, you're just doing boxes and arrows. Like you just want to show that you're mapping out space, you're mapping out very basic layout and interaction elements. As you continue down the path and you increase fidelity, then you can add to that. But in the beginning, a lot of clients that I've interacted with are kind of a little put off when they get their initial wireframes and it is just a couple boxes. What is this?

Presenter: It's like, "Trust me. This gets better."

Presenter: Yeah, and--
~~~
Presenter: You call yourself a designer?

Presenter: This took you two weeks? What?

Presenter: We did really good research.
~~~

Presenter: I promise you this means something.

Presenter: That's right.

Presenter: And then you continue through the process increasing the fidelity, getting to the point where you're starting to build click-through prototypes to make sure that the interaction that you're seeing on the page is reinforced by the actions that the user might complete. I found that task based user testing is really helpful for that. You present a user with a task, say, "I want you to complete this element," and see how long it takes them to do it, or if they can do it at all.

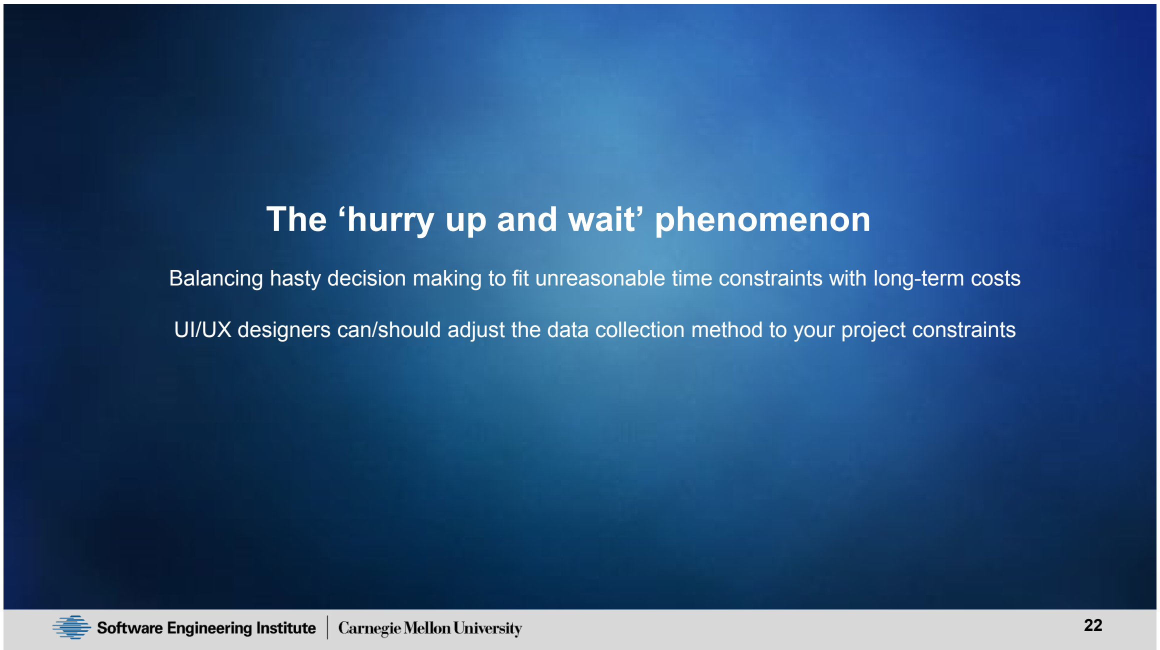
Presenter: And accuracy and roots. And it's even more exciting-- I was just telling the group this before the webinar started, it's more exciting, though, to-- when you're doing that kind of user testing where you have a click-through prototype and you have the developers sitting behind in another room watching through the live camera feed as the person is clicking through the interface. And they're like, "Rats, that's not what's supposed to happen. And that's not how I intended that to go." And it's fun. I mean it's kind of funny to listen to them complain.

But they get-- remember, some of your designers or some of your developers are very black and white thinking. And you're thinking in abstraction and black and white. And now, you've transferred that to an interface. And they're thinking black and white. And it's like no, no, no, it's an abstraction that they're thinking. So, you're trying to bridge the two worlds. The user has a different perspective than developers. And that's a hard one to teach unless you have them live on the wire.

Presenter: Absolutely, it's excellent to get that immediate feedback. And I think that that's, at least for me as a user experience designer, that's been one of the most valuable and interesting parts of my job is being able to put my work in front of somebody and actually get really insightful and quality feedback that helps me grow as an individual as well.

Presenter: That's right.
There is one caveat, I know we didn't cover this.

The ‘hurry up and wait’ phenomenon



The ‘hurry up and wait’ phenomenon

Balancing hasty decision making to fit unreasonable time constraints with long-term costs

UI/UX designers can/should adjust the data collection method to your project constraints

Software Engineering Institute | Carnegie Mellon University 22

**022 But in the lifecycle. It's always a good idea to have the lifecycle-- UI/UX help anywhere in the lifecycle. But there is-- I've found it very challenging to come in at the end of the development and right before you launch the new version or the new product and you are going to evaluate using a standard UI task based test and evaluation. They will complain because oftentimes, they haven't done any UI/UX work at all. And now, you're catching all of the problems. And you have a list that's five miles long. And it goes right into their backlog. And they're like, "No, no, no, we have to launch this in two weeks." And you're like, "You've got six months more worth of effort here." And they just, they hate you. So, it's a hard place to be.

So, often, I weight the pros and cons of helping them at that point in time because you want to do the right thing, but you don't want to make them mad at you and never have another UI/UX designer hired.

Presenter: Oh yeah, absolutely. But at least for me, I mean advocating on the part of the user, I think that there's-- oftentimes, there's the concept of human centered design versus user centered design, adding in the needs of the business as a component of the decision making process. In some situations, that can be difficult. You have situations where the organization has very different needs than the end user might. Or at least they have different interests from a profit perspective, so kind of balancing those elements. But the main point being documenting those, understanding that this is what we saw with the population.

Organizationally, the organization needs something different. That's fine. You are designing something for that organization. So, if it doesn't fit their needs, it doesn't fit a large percentage of the population's needs. It's their money.

Presenter: That's the hard part is like you're designing for the organization not the end user. But it's the culture and the check is coming from somewhere.

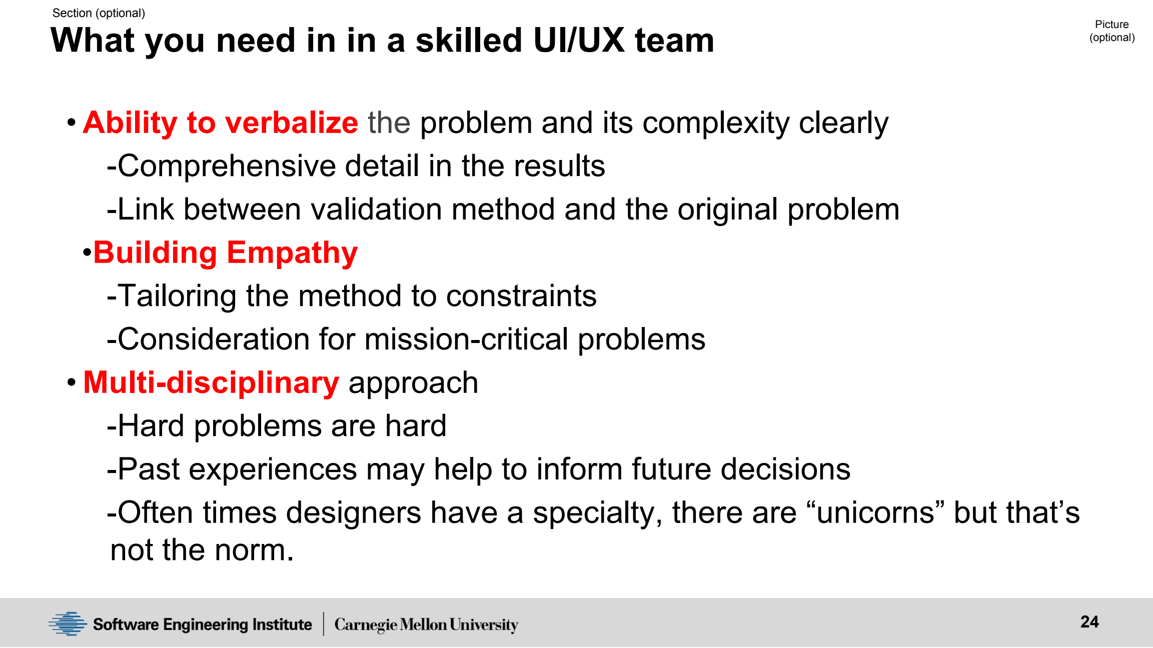
Presenter: Yeah, but you need to document it. You need to document it and say this is what we observed. Yes. This is what we observed. This is what we saw. This is something we observed in this particular aspect. This is kind of where it met in the middle. And I think that that's perfectly acceptable. And I think that it's perfect understandable. But I do think that it's a valuable thing to kind of have all of those elements documented and having all of those-- the artifacts of the process are what kind of structure that.

Presenter: That's right. That's right. Oh, my favorite slide. This is the hurry up and wait phenomenon. And I know we have five minutes left. So, I'm going to hurry up through this hurry up and wait. The hurry up and wait phenomenon is like you hurry up to get through-- you quickly generate the problem and the requirements. And you just slap a tool at it. And then you-- because you're hasty, you're in a hurry. This world that we live in in the digital age, we've allowed people to be really quick thinkers but also maybe not the best decision makers. And so, they quickly make a decision. And then the waiting part is, "Oh shoot, this doesn't fit at all. My people aren't using this tool. And now, I have to redo this whole process."

So, the point of this is, is make sure that you balance hasty decision making with the best interests of the customer. And you try to educate the customer as you're going through it and saying, "From my personal experience, the return on investment is really going to go down if you have me at the end. It's really going to go down. You're going to increase your bug testing timelines," etc. So, you really want to make sure you have an honest conversation with these people.

Presenter: Sure, and--

What you need in in a skilled UI/UX team



Section (optional)

What you need in in a skilled UI/UX team

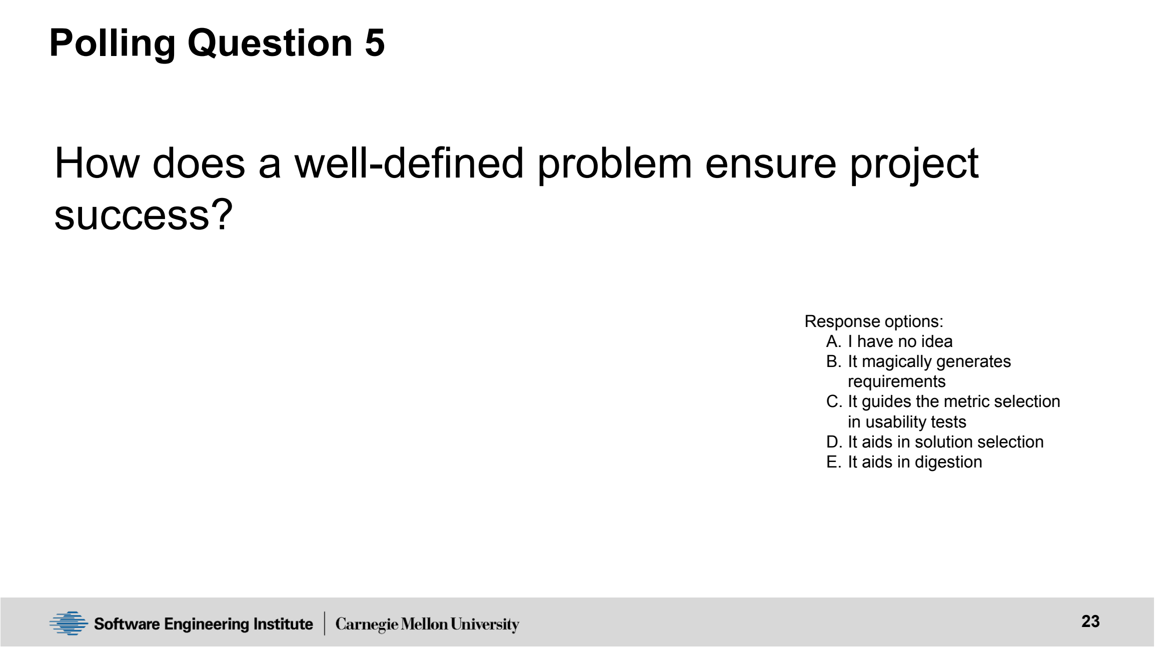
Picture (optional)

- **Ability to verbalize** the problem and its complexity clearly
  - -Comprehensive detail in the results
  - -Link between validation method and the original problem
- **Building Empathy**
  - -Tailoring the method to constraints
  - -Consideration for mission-critical problems
- **Multi-disciplinary** approach
  - -Hard problems are hard
  - -Past experiences may help to inform future decisions
  - -Often times designers have a specialty, there are “unicorns” but that’s not the norm.

Software Engineering Institute | Carnegie Mellon University 24

**024 Presenter: Oh, we have one more polling question.

Polling Question 5



Polling Question 5

How does a well-defined problem ensure project success?

Response options:

A. I have no idea
B. It magically generates requirements
C. It guides the metric selection in usability tests
D. It aids in solution selection
E. It aids in digestion

Software Engineering Institute | Carnegie Mellon University

23

**023 Whoops.

Presenter: So, this one, we'll just post this one up there quickly just to wrap up.

Presenter: Oh, this is the magic one.

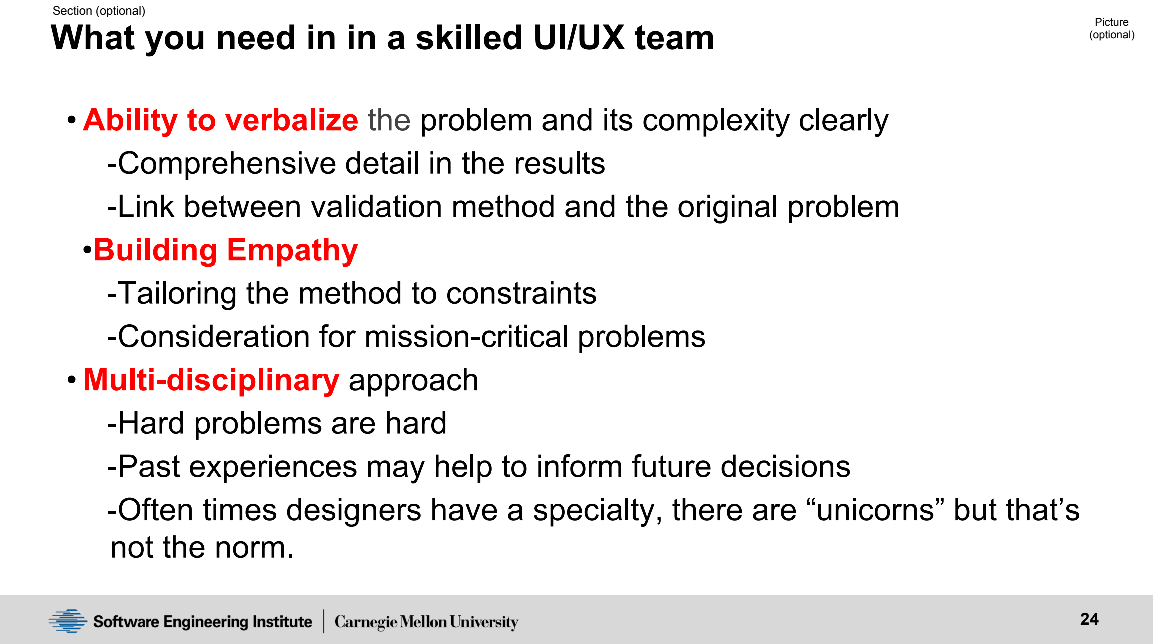
Presenter: How does a well-defined problem ensure project success? You'll see that on your screen now. While you're voting, just a couple questions are coming in the chat as well asking about if this is being archived for later viewing. It is. It will be ready as soon as early as tonight or definitely by tomorrow morning. You just log back in with your same credentials today. And you'll be able to access the recording of that and the slides and all the information there. And also, we just request that you do fill out the survey upon exiting todays' webinar because we'd love to get your feedback. Also, there were a couple questions in there about what other topics you'd like to see in the area of UI/UX.

So, let's see the results real quick. And keep going. It's seven percent it naturally generates requirements, twenty-seven percent it guides the metric selection in usability tests. And sixty-seven percent, it aids in solution selection.

Presenter: Yeah, C is the hot one. And very quickly, why is that often we see testing. They say, "Oh, bug testing verifies that the tool works." Okay, but did you solve the problem. So, we're actually linking it back by choosing the right metric to link it back to the problem. So, we're demonstrating very clearly to the customer that we actually did solve part of the problem, or how well we solved it.

Presenter: Absolutely, and--

What you need in in a skilled UI/UX team



Section (optional)

What you need in in a skilled UI/UX team

Picture (optional)

- **Ability to verbalize** the problem and its complexity clearly
  - -Comprehensive detail in the results
  - -Link between validation method and the original problem
- **Building Empathy**
  - -Tailoring the method to constraints
  - -Consideration for mission-critical problems
- **Multi-disciplinary** approach
  - -Hard problems are hard
  - -Past experiences may help to inform future decisions
  - -Often times designers have a specialty, there are “unicorns” but that’s not the norm.

Software Engineering Institute | Carnegie Mellon University 24

**024 Presenter: Oh yeah, the last couple of things--

Presenter: Quickly, just to cover in the last couple minutes here, when you're building your user experience team, what are you looking for? To touch on it very quickly, the ability to verbalize. There are many complex problems within the user experience and development design environment. Being able to provide complete-- look at me stuttering while I'm trying to explain it, ironic.

Presenter: Ten.

Presenter: Yeah. Comprehensive details in the results and a link between their validation method and the original problem, how did they come to that conclusion.

Presenter: Yeah, how did they do that is always the key.

Presenter: Yes, absolutely, building empathy, tailoring the method to the constraints and consideration for all the mission critical problems.

Also, having a multidisciplinary approach, hard problems are hard. That's kind of just the way it is.

Presenter: It just is, yeah.

Presenter: And past experiences may help to inform future decisions. So, really understand how things have gone in the past, how things-- might define how things have to happen in the future.

Also, being understanding that designers have a specialty. There are unicorns. Those are the designers, the developers, everything people. They exist. But they are rare. And it's not often that the unicorns are a master of everything. They are--

Presenter: Just of what they know.

Presenter: Yeah.

Presenter: Right, so that's why a diversified team is important.

Presenter: Absolutely, so--

Presenter: Of unicorns.

Presenter: Yeah, if you find one, get a hold of them, hold on tight. But understand that that's having a very,

very diverse skill set that they can deep dive into every section could be a little difficult at times.

Presenter: That's right, absolutely.

Presenter: Great presentation. We know it's two thirty, folks. But if you guys give another minute, maybe we can just--

Presenter: Yeah, we can go through a couple of these great questions.

Presenter: We'll just take a minute. So, if there's something-- there's a couple that were in the chat from earlier. If it's something that's not relevant, maybe we just say pass. And we'll address it later. But one came in earlier asking, "Can you compare and contrast with Hutchin's Distributed Cognition?" And if it's not something relevant, like I said, we'll pass. And we can move on to another question.

Presenter: That's not quite my area of expertise. And I'll be honest, I don't know what Hutchin's-- I know what distributed cognition is. But I don't know what that particular model says. So, I apologize.

Presenter: Sure. So, we'll go on to the next one. "What are the top three UX research methods you recommend for a new team getting started with UX, if there is such methods?"

Presenter: Oh, that's right. Well, that depends on timing. So, often we choose quick and dirty methods that still give you some quality work. Like heuristic evaluation, cognitive walkthroughing can get a lot of problems solved immediately if you have a real big time constraint.

Presenter: Yeah, starting off with a heuristic evaluation following Nielson's heuristics is a really great place to get started. You can also, depending on the type of content that you're creating--

Presenter: Testing is really critical. The iterative design and tests, if you can have a lab-- I know that's costly sometimes, but not all the time. You just buy Camtasia-- or not Camtasia, but Morae-- Morae Studio, you can quickly do some really off the cuff testing and really-- remotely, you can even remote desktop in to somebody, some user's station, and then you can really do-- you can do some effective understanding of how well that design is actually working.

Presenter: Less expensive, just doing card sorting when you're doing information architecture and the type of concepts that you can glean from there can be very insightful, not very expensive, not a super involved process. But there's--

Presenter: If I had six pages of writing I could give you, I would because there's a lot more to answer that, but--

Presenter: And let's wrap up with, "Almost every piece of software is moving to the cloud. Do you think this change needs to change in UX/UI thinking?" I'm sure the obvious answer is yes. But I wanted to hear your experience where you had to redesign the UI.

Presenter: For moving to the cloud, mostly in databasing, but--

Presenter: Well, yeah I mean there's definitely a lot of software if you take like Microsoft's moving Office to the cloud. I mean when we were working on our presentation, we were working using Google's tools. There-- as long as you're connected to the Internet, if you live in an environment that that is acceptable, it's--

Presenter: That's the key. Yeah, really.

Presenter: It's establishing the fallback scenarios that are-- that can be difficult. Living here in western Pennsylvania, I did a study of adoption of Internet within the home and Internet speeds and found this area is a very unique environment. We are very, very far behind relative to the rest of the United States as a whole. There are many places that people don't have access to high speed Internet or are not interested in it in the first place. So, kind of--

Presenter: And those apps just don't render out.

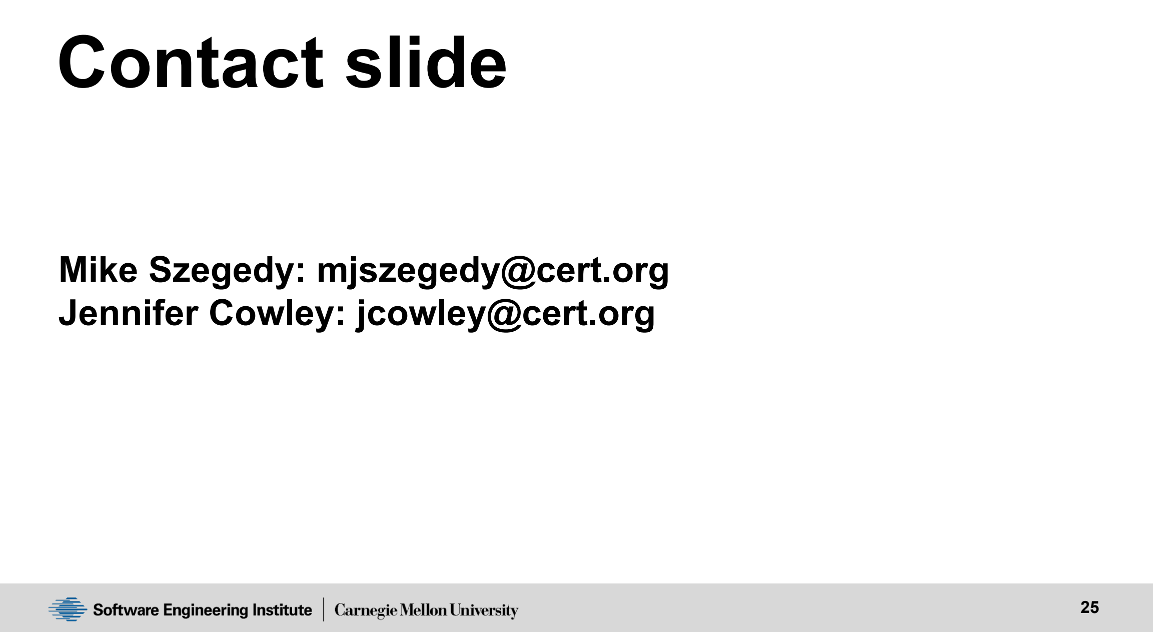
Presenter: Right, so as long as the system is designed to support the worst case scenario where you don't have that type of functionality, and you can still complete the task as desired and then have online sync up whenever you are back in within that environment, I don't see why most apps couldn't mainly function off of an online environment.

Presenter: Yeah, the online key, that's a real lynch pin because there are some-- there's a lot of distractibility now if you want to look at the clinical psychology literature. There's a lot of distractibility that comes with connecting to the Internet. People know they're connected to the Internet, and they're getting-- they're self-distracting. And so, if you have a job that you can actually take it off the Internet and put it on a thin client that works, but if you have to force them to that, you have to make sure you're cross comparing the cost-benefit of that move if we're a really deeply technical position that has to spend a lot of hours thinking.

Presenter: Right, Michael, Jennifer, thank you for your talk, excellent presentation. Folks, that's going to wrap up our talk today. So, we thank you for spending the afternoon with us.

Our next webinar will be October-- Monday, October 10th.

Contact slide



Contact slide

Mike Szegedy: mjszegedy@cert.org
Jennifer Cowley: jcowley@cert.org

Software Engineering Institute | Carnegie Mellon University

25

**025 And the topic will be building analytics for network flow records with Tim Schimel and Matthew Heckathorne. And everyone signed up today will get an invite to that. Thanks, everyone. Have a great day.

SEI WEBINAR SERIES | Keeping you informed of the latest solutions

SEI WEBINAR SERIES | Keeping you informed of the latest solutions

Software Engineering Institute | Carnegie Mellon University