

Usability Testing

Goals of today

- Learn why developers perform usability testing
- Learn the basics of how usability testing is performed
- Design and facilitate a usability test for your project
- Participate in a usability test of someone else's project
- Leave with at least one usability-related problem you'd like to **fix during your last sprint**

What is usability testing?

“Watching people try to use what you’re creating/designing/building, with the intention of (a) making it easier for people to use and (b) proving that it is easy to use”

- Steve Krug, *Rocket Surgery Made Easy: The Do-It-Yourself Guide to Finding and Fixing Usability Problems*

5 Components of Usability

- **Learnability:** How easy is it for users to accomplish basic tasks the first time they encounter the design?
- **Efficiency:** Once users have learned the design, how quickly can they perform tasks?
- **Memorability:** When users return to the design after a period of not using it, how easily can they reestablish proficiency?
- **Errors:** How many errors do users make, how severe are these errors, and how easily can they recover from the errors?
- **Satisfaction:** How pleasant is it to use the design?

Nielsen (2012). Usability 101: Introduction to Usability
<https://www.nngroup.com/articles/usability-101-introduction-to-usability/>

I want to largely focus on learnability today. However, you've all seen demos of each other's applications, so in practice this may be more like "memorability".

Types of Usability Testing

- **Quantitative:** try to *prove something* about your application by *measuring things*
 - Answer questions such as:
 - Is the new version **better** than the last version?
 - Are people able to **quickly** complete important tasks in the application?
 - Requires a scientific approach
- **Qualitative:** gain *insights* into how you might be able to improve your application
 - Participants try to perform a task and **think out loud** while doing it
 - Requires fewer participants and testing structure can be more flexible

Krug (2010), Rocket Surgery Made Easy: The Do-It-Yourself Guide to Finding and Fixing Usability Problems

We'll do a quick qualitative test with a few participants. In this class, our main goal for usability testing will be to reveal pain points that might not be obvious to the developers of the application.

Step-by-Step Guide

- Recruit **representative users**
- Ask those users to perform a set of **representative tasks**
- **Observe** what the users do and where they have difficulties with the user interface
- Debrief and determine **what you will do** to address usability problems

Nielsen (2012). Usability 101: Introduction to Usability
<https://www.nngroup.com/articles/usability-101-introduction-to-usability/>

Representative users: As developers, you know how the application is supposed to work – you built it! Try to find users who match your target users for testing. Today, you’ll use your classmates as representative users. However, it’s worth thinking about how they might differ from “real” representative users!

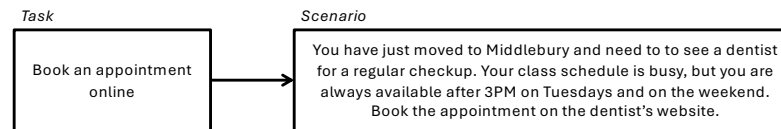
Representative tasks: we want to consider the most important tasks for our application (more on the next slide)

Observation: take notes as you perform the test so that you are able to reflect on what happens

Debrief: in addition to coming up with a set of problems, you should have actions that you’ll take to address those problems

Defining Tasks & Scenarios

1. Come up with a list of the most important things that a user needs to be able to do on your site (**tasks**)
2. Convert those tasks to scripts that give context on how they will perform the tasks (**scenarios**)



Krug (2010), Rocket Surgery Made Easy: The Do-It-Yourself Guide to Finding and Fixing Usability Problems

Tasks are more general, while scenarios fill in the holes to help the user perform the task as you intend

Creating High-Quality Scenarios

Make the Scenario Realistic

- **Task:** Browse product offerings and purchase an item.
- **Poor scenario:** *Purchase a pair of orange Nike running shoes.*
- **Better scenario:** *Buy a pair of shoes for less than \$40.*

Make the Scenario Actionable

- **Task:** Find movie and show times.
- **Poor scenario:** *You want to see a movie Sunday afternoon. Go to www.fandango.com and tell me where you'd click next.*
- **Better scenario:** *Use www.fandago.com to find a movie you'd be interested in seeing on Sunday afternoon.*

Avoid Giving Clues and Describing the Steps

- **Task:** Look up grades.
- **Poor scenario:** *You want to see the results of your midterm exams. Go to the website, sign in, and tell me where you would click to get your transcript.*
- **Better scenario:** *Look up the results of your midterm exams.*

McCloskey (2014). Turn User Goals into Task Scenarios for Usability Testing
<https://www.nngroup.com/articles/task-scenarios-usability-testing/>

Setting Up the Test and Preparing Participants

- Make sure that the participant knows that **the product** is being tested, not **the participant**
- Give the participant written instructions describing the task
 - Read the task out loud to the participant to make sure they aren't missing anything
- Give the participant one task at a time

In our user tests, we'll have printed out tasks and scenarios; you don't need to worry about prepping the participant, because you've all been here in this lecture.

During the Test

- Have the participant “**think aloud**”
 - What are they doing?
 - Why are they doing it?
- **Don’t help** the participant!
 - Remember, you wouldn’t be in the room with a real user!
 - Use your judgement – you may be able to answer some clarifying questions
- Take detailed **notes**
 - What steps did the user take? Where did they click?
 - How long did it take them?
- When to **stop** a task?
 - Keep an eye on the time
 - Make sure you’re learning something

Krug (2010), Rocket Surgery Made Easy: The Do-It-Yourself Guide to Finding and Fixing Usability Problems
<https://hci.stanford.edu/courses/cs147/2022/wi/lectures/15-usability-testing.pdf>

You should ensure that the user is “thinking out loud” to help you to understand their actions and how they relate to your task. Because you want to mimic a real-life scenario (where the developer is not sitting with the user), you generally shouldn’t help the user and push them towards taking any actions that they wouldn’t otherwise take. You should take detailed notes, and try to make sure that things stay on track in terms of time.

Debriefing

- Your debriefing meeting should lead to two lists:
 - The most serious usability problems that were uncovered in your site
 - The usability problems that you intend to fix before the next round of usability testing
- One way to start: go around and have everyone who observed the test list the 3 most serious problems that they saw

Adapted from Krug (2010), Rocket Surgery Made Easy: The Do-It-Yourself Guide to Finding and Fixing Usability Problems

The final step is to debrief and identify what you can do going forward to fix some of the usability problems in your application! The prompt at the bottom can help to get you started.

Example: Transit Planning

You are a student at Middlebury who is starting an internship at UVM medical center. You do not currently have a car, and decide to take the bus to Burlington.

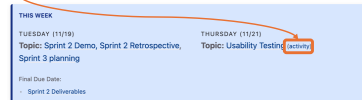
You need to get to the medical center bus stop by 8:30AM. Figure out when you need to be at the Academy Street bus stop in Middlebury to make it on time.

<https://www.trivalleytransit.org/>

We'll have a volunteer try to perform this task on the tri-valley transit website. There are two options: one takes you to google maps (and is further down the page), while the other requires identifying the correct route(s) and looking at their schedules.

Usability Testing Activity

Please see the **details** on the course website



1. Define Tasks and Scenarios
2. Prepare Your Environment
3. Conduct Your Test
4. Debrief

The final 45 minutes of class are dedicated to designing, facilitating, and participating in a usability test for your projects.

Standup Meeting Prompts

Each team member should discuss:

- What they did since the last class to help the team meet the Sprint Goal
- What they plan to do between now and the next class
- Any impediments that will prevent the team from meeting the Sprint Goal