AARHUS
UNIVERSITET

USER INTERFACES FOR EMBEDDED SYSTEMS

Lecture 10: Performance Metrics

Stefan Wagner
sw@eng.au.dk

AGENDA

1. Systematic usability evaluation with Metrics
2. Definition of Metrics
3. Different types of Metrics
4. Choosing the right Metric
5. Exercise

SYSTEMATIC USABILITY EVALUATION

- If we want to learn whether design prototype A or B is best
  - We need a way to say A > B, B > A, or A = B
  - To do so, we need our data to be comparable: “Metrics”
  - Our methods must be valid and reliable
  - Thus, we need our evaluation to be relevant, representable, systematic, and reproducible

- Thus, we need to define a series of experiments to support our arguments – with sufficient data to give us the statistical power we need (number of test users, number of test cases)

EXERCISE

- Take 10 minutes and discuss what is actually relevant to test in your prototypes?
- Is it enough with a simple test?
- Do you need to test with the same participant many times?
- Should the participant receive training?
- Should you redo the tests over longer periods of time?

METRICS: WHAT?

- What are metrics?
  - A way of measuring or evaluating a particular phenomenon
- Metrics should be
  - General
  - Standardized
  - Consistent
  - Reliable
  - Observable
  - Quantifiable
- Think of: Length (the meter), mass (the gram) etc.
  - Système international d'unités (SI)

METRICS: WHY?

- Improve your products
  - Help design decisions
  - Estimate size and magnitude of issues
  - Compare designs and design iterations
  - Reveal patterns
- Improve your project reports
  - Structure experiment designs
  - Test your hypotheses
  - Stronger conclusions

Table 3.1 Ten Common Usability Study Scenarios and Their Most Appropriate Metrics

Usability Study Scenario	Task Success	Task Time	Errors	Efficiency	Learn-ability	Issues-Based Metrics	Self-Reported Metrics	Behavioral and Physiological Metrics	Combined and Comparative Metrics	Live Website Metrics	Card-Sorting Data
1. Completing a transaction	X			X		X	X			X	
2. Comparing products	X			X			X		X		
3. Evaluating frequent use of the same product	X	X		X	X		X				
4. Evaluating navigation and/or information architecture	X		X	X							X
5. Increasing awareness							X	X		X	
6. Problem discovery						X	X				
7. Maximizing usability for a critical product	X		X	X							
8. Creating an overall positive user experience							X	X			
9. Evaluating the impact of subtle changes										X	
10. Comparing alternative designs	X	X				X	X		X		

Different scenarios require different metrics

[T&A]

USER GOALS: PERFORMANCE

- Performance is about what the user actually **does** when interacting with the product
- Measuring degree to which users can successfully accomplish tasks:
  - How long does it take to perform a task?
  - How much effort does it take to perform a task, e.g.
    - mouse clicks or cognitive load
  - Number of errors committed
  - Learnability: How long does it take to become proficient?
- Users must be able to perform key tasks successfully
- We will return to performance measures later

USER GOALS: SATISFACTION

- Satisfaction is about what the user **says** or **thinks** when interacting with the product
- The user might report that the product, e.g.
  - is easy to use
  - is confusing to use
  - exceeded expectations
  - is visually appealing
  - is untrustworthy
- If the user is not satisfied, he is unlikely to, e.g. buy the product or spend time on the website
- Satisfaction is a **self-reported** metric. They will be reviewed later

CHOOSING THE RIGHT METRICS

- Most usability studies have unique qualities
- When choosing your metrics, consider, e.g.
  - Study goals
  - User goals
  - Time-line, resources, budget, etc.
  - Possibilities of data acquisition
  - Data types
  - Explore your raw data and develop **new** metrics if needed

PERFORMANCE METRICS (PM)

- Task success
  - How effectively users are able to complete set of tasks
- Time-on-task
  - How much time required to complete a task
- Errors
  - How many errors are made while completing a task
- Efficiency
  - Effort to complete a task, e.g. number of mouse clicks
- Learnability
  - How performance changes over time, i.e. learning as we go

Performance metrics are good at identifying **what** is wrong (**effect**), but not **why** (**cause**)

PM: TASK SUCCESS

- Binary success
  - True or false: Either succeed or fail
  - Frequency of users succeeding per task
- Levels of success
  - Extent of task completion: numerical value or percentage to each level
  - Experience in completing the task: Easy, medium, struggling
  - Way of task completion: Optimal way or alternative way
- User segmentation
  - Frequency of use, previous experience, expertise, age, gender, etc.
- Issues
  - When to stop, i.e. how long time to give the user
  - Define success criteria, task end state

BINARY SUCCESS

B14 =AVERAGE(B2:B13)

	A	B	C	D	E	F
1	**Participant**	**Task 1**	**Task 2**	**Task 3**	**Task 4**	**Task 5**
2	P1	1	0	1	0	0
3	P2	1	0	1	0	1
4	P3	1	1	1	1	1
5	P4	1	1	1	1	1
6	P5	0	0	1	1	1
7	P6	1	0	0	1	1
8	P7	0	1	1	1	1
9	P8	0	0	1	1	0
10	P9	1	0	1	0	1
11	P10	1	1	1	1	1
12	P11	0	1	1	1	1
13	P12	1	0	1	1	1
14	**Average**	**67%**	**42%**	**92%**	**75%**	**83%**
15	**Confidence Interval (95%)**	28%	22%	29%	29%	29%
16						

0 = Task failure

1 = Task success

=AVERAGE(F2:F13)

Calculated based on binomial distribution

USING BINARY SUCCESS

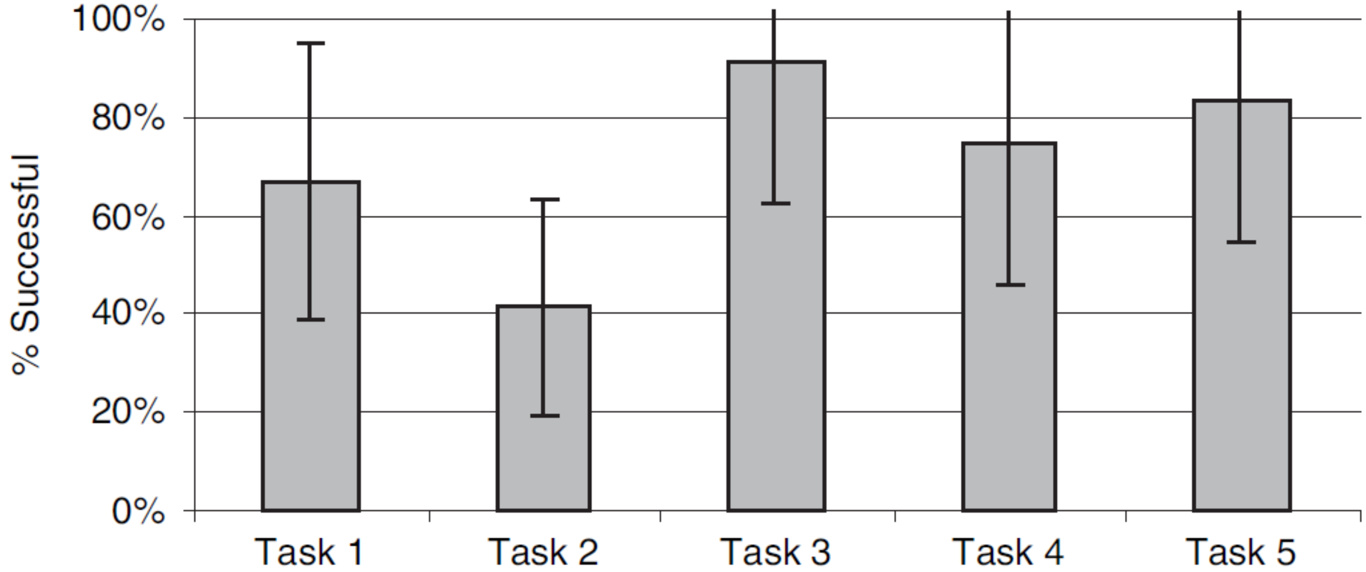

FIGURE 4.2

An example of how to present binary success data for individual tasks. The error bars represent the 95 percent confidence interval based on a binomial distribution.

VISUALIZING DIFFERENT PROBLEMS

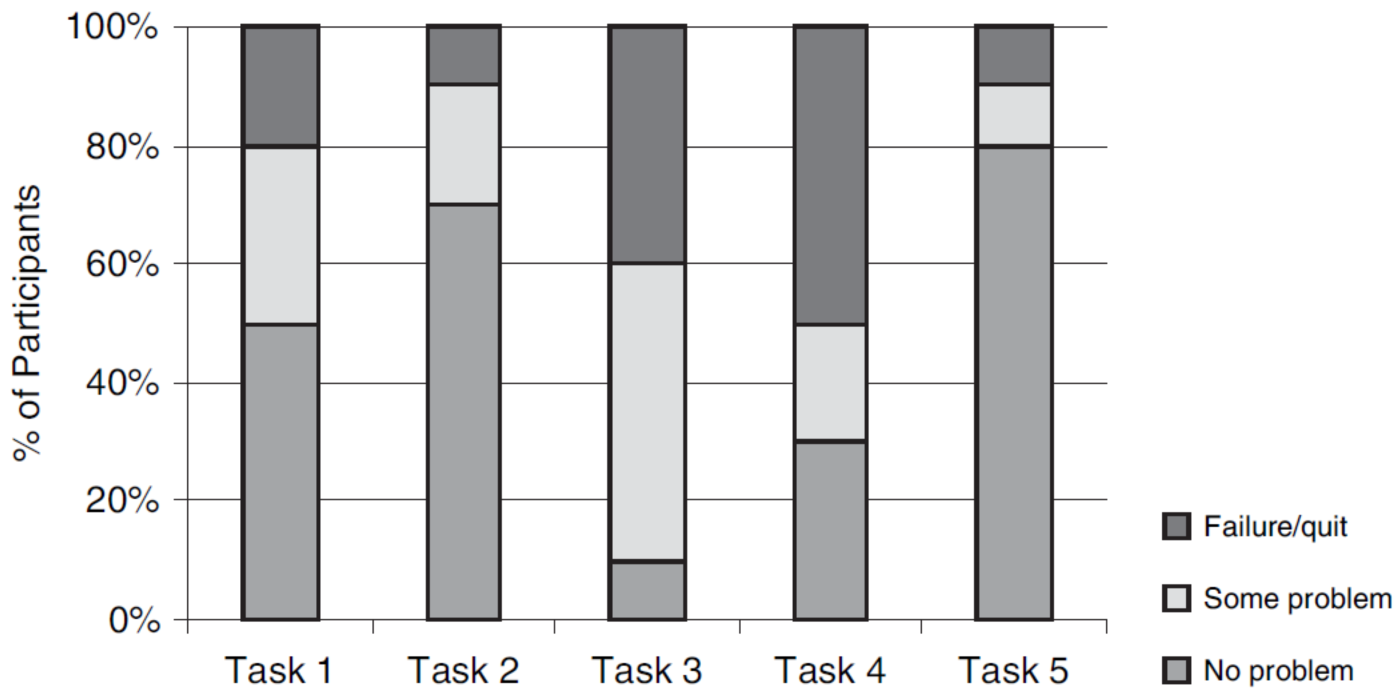

FIGURE 4.4

Stacked bar chart showing different levels of success based on task completion.

PM: TIME-ON-TASK (1/2)

- Time-on-task = $\Delta t = t_{Task\text{-}End} - t_{Task\text{-}Begin}$
- Also known as “task completion time” or “task time”
- Very important Metric for:
  - Frequently carried out tasks (expert users with many similar tasks)
  - Multi-user scenarios with peak stress (kiosk systems)
- Timing: Automatic or manual
  - Auto timestamps via tools or code instrumentation
  - Manual timestamps via stop watch
- Average amount of time spent: mean or median
- Good for comparing effectiveness of design A vs B

MICRO EXERCISE (5 MINUTES)

- Discuss with your neighbor:
- How can we automatically record Time-on-Task?
- How would you store it – and how would you get it into e.g. Excel?
- What happens if you use the Think Aloud Protocol with Time-on-Task?

PM: TIME-ON-TASK (2/2)

- Issues
  - When to stop the clock?
  - Use data only from successful tasks?
  - Is experiment influencing time, e.g. think-aloud protocol
  - Should the users know they are being timed?
    - Influences exploratory behavior, nervousness, etc.
  - Filter outliers? What if user is interrupted?
  - Apply training before starting?

PM: ERRORS (1/2)

- Measuring errors may be important if
  - They cause loss of efficiency
  - They result in significant extra costs
  - They result in task failures
- Single or Multiple error opportunities
- Counting errors
  - Frequency / percentage of errors per task
  - Percentage of users that makes an error in each task
  - Define acceptable error levels, e.g.
    - 20% of tasks had error rate above acceptable level (10%)

PM: ERRORS (2/2)

- Errors are **not** issues
  - Issues are the **cause** of errors (**effect**)
- Issues
  - Need to define correct actions to know when errors occur
  - May need to classify types of errors
  - Do not double count errors

PM: EFFICIENCY (1/2)

- Amount of **effort** required to complete a task
  - Physical load
  - Cognitive load
- Not just **time**, but also **actions**, e.g.
  - Number of steps/actions required to complete a task
  - Average no. of actions per task for each participant
  - Number of mouse clicks and/or key presses
- **Combine** tasks success, time, and actions
  - When completing a task successfully:
    - How much time spend? No. of actions performed
  - When failing a task
    - Time spend / actions performed before giving up?

PM: EFFICIENCY (2/2)

- Issues
  - Define which actions to count
  - Define when to start and end counting actions
  - Cognitive load
    - Mental actions can be hard to measure
    - Cognitive load is hard to measure
  - Physical load
    - Mouse clicks and key presses are easy to measure
    - Additional mouse clicks done to explore the interface?

EXAMPLE: WEB "LOSTNESS"

N: The number of *different* web pages visited while performing the task

S: The *total* number of pages visited while performing the task, counting revisits to the same page

R: The *minimum* (optimum) number of pages that must be visited to accomplish the task

Lostness, *L*, is then calculated using the following formula:

$$L = \mathrm{sqrt}[(N/S - 1)^2 + (R/N - 1)^2]$$

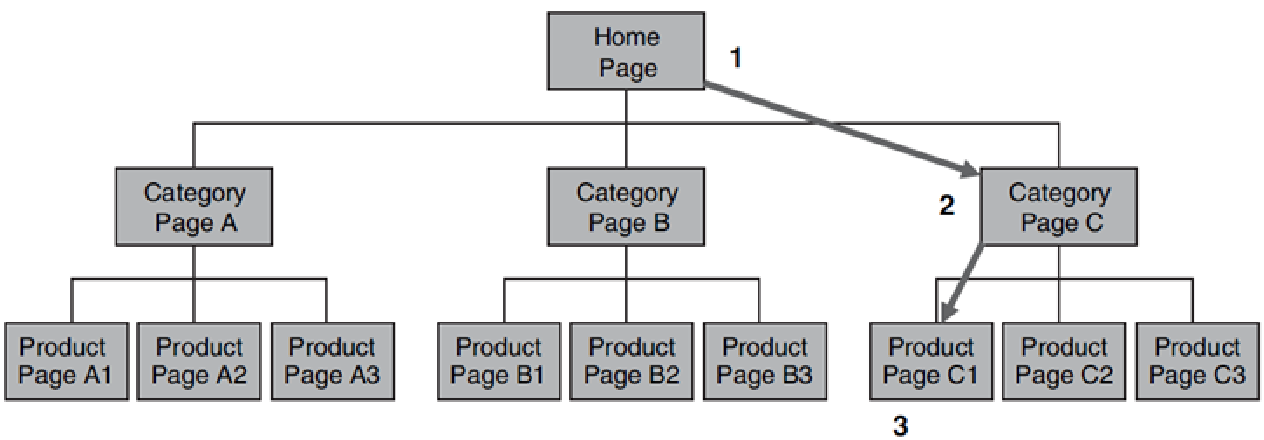

FIGURE 4.9

Optimum number of steps (three) to accomplish a task that involves finding a target item on Product Page C1 starting from the homepage.

EXAMPLE WEB "LOSTNESS" (2/2)

› Can be measured with standard statistics web module
› Perfect Lostness L = 0

$$N = 6$$
$$S = 8$$
$$R = 3$$
$$L = \text{sqrt}[(6/8 - 1)^2 + (3/6 - 1)^2] = 0.56$$

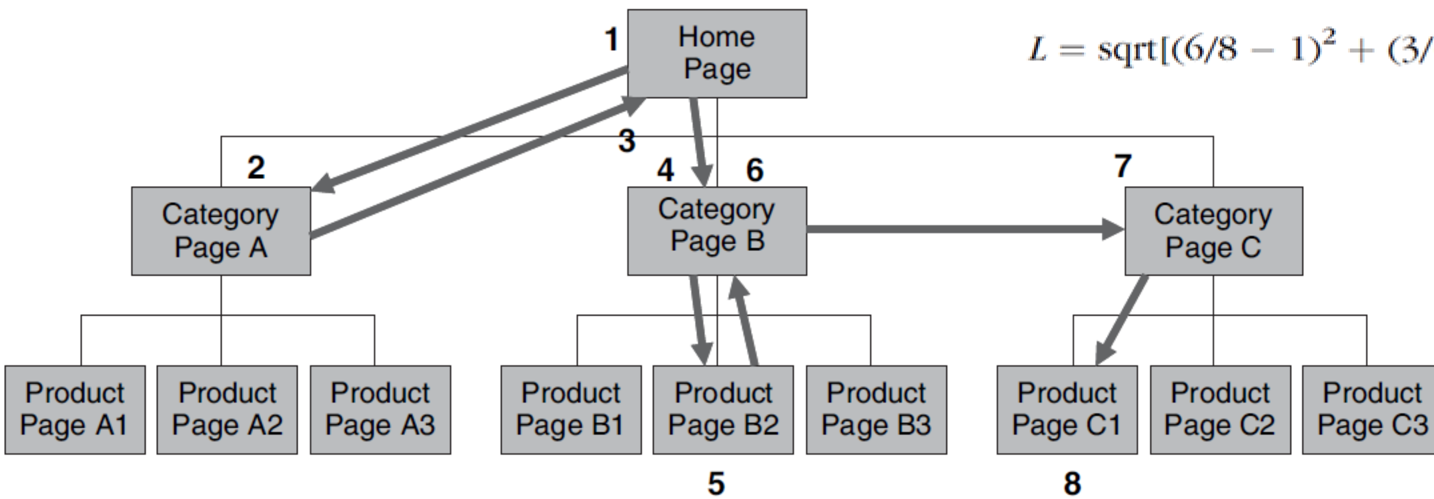

FIGURE 4.10

Actual number of steps a participant took in getting to the target item on Product Page C1. Note that each revisit to the same page is counted, giving a total of eight steps.

PM: LEARNABILITY (1/2)

- Develop a metric that is a **function** of time and trials
  - Success rate
  - Time-on-task
  - Number of errors, etc.
- Demonstrate a **learning curve** within or between sessions
  - Interpret the curve and look for oddities
- Plot as graph and look for
  - When and if it flattens out, i.e. no more learning occurs
    - Time spend / trials done before max performance reached
  - Difference between, e.g.
    - Learning curve of experts vs. novices

PM: LEARNABILITY (2/2)

- Issues
  - - How to define a trial
    - When does trial begin and end?
- What if interaction and learning is continuous?
- How much time should pass between trials?
  - - How many trials to include?
    - Minimum two, often more

VISUALIZING LEARNABILITY

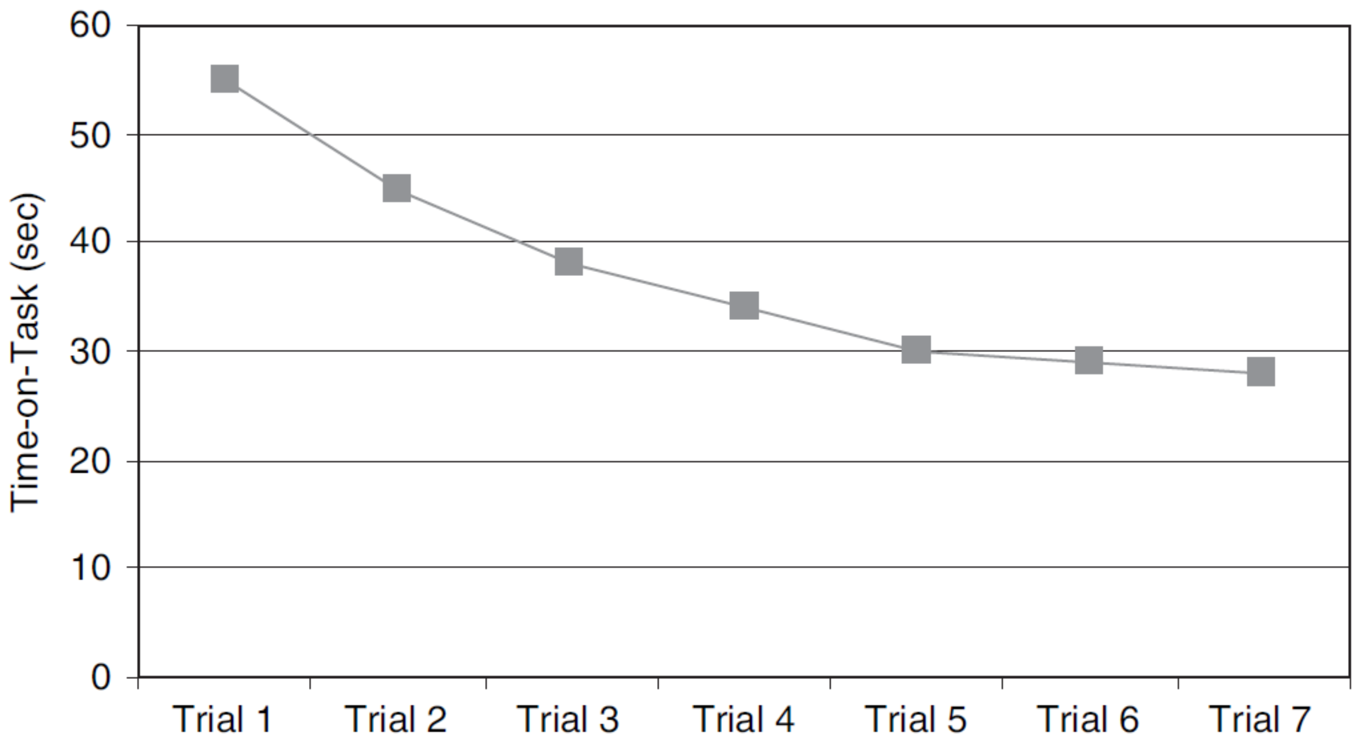

FIGURE 4.13

An example of how to present learnability data based on time-on-task.

COMPARISONS

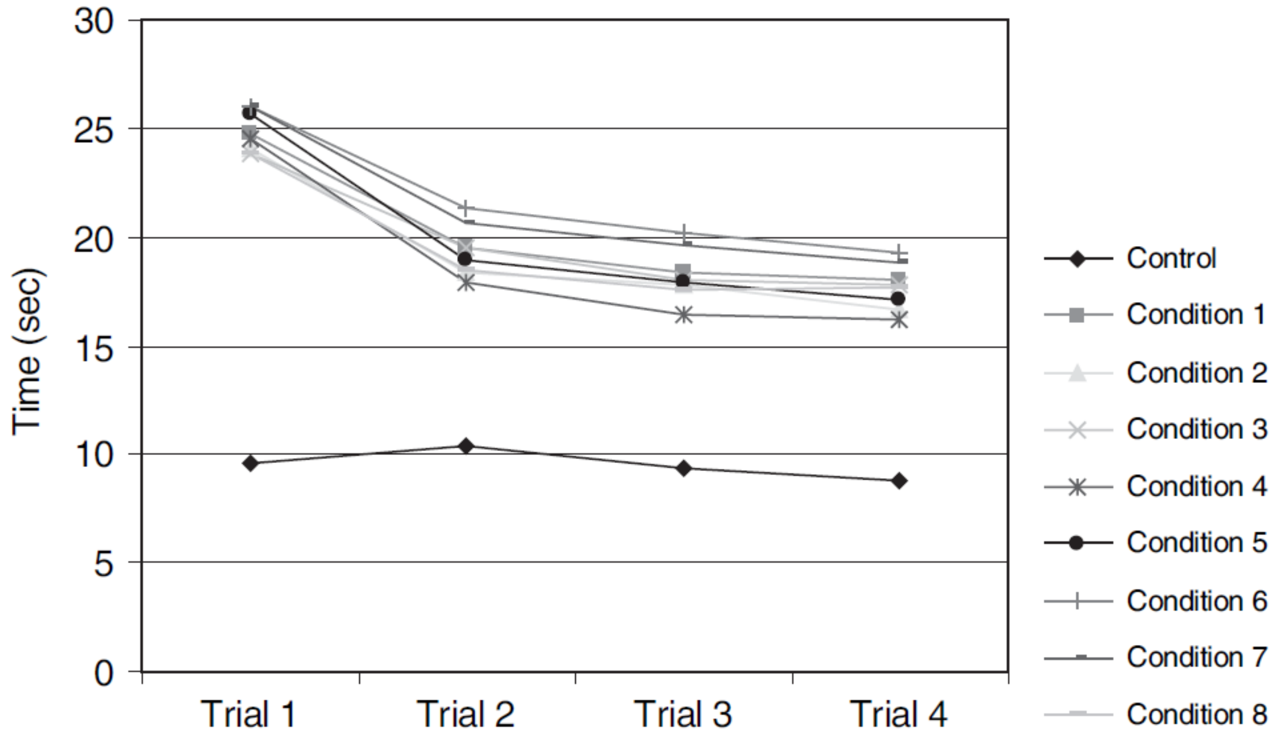

FIGURE 4.14

Looking at the learnability of different types of on-screen keyboards.

EXERCISE

- How will you be able to say prototype A > B
- Create a test plan for measuring performance metrics
- In a document
- - Discuss which tasks you plan to compare
- - Discuss what knowledge you plan to gain (your research question)
- - Discuss which metrics you plan to use (focus on performance metrics)
- - Discuss how many participants you need, and how much time is needed

- Recruit test users and do the evaluation!

REFERENCES

[T&A] Tullis and Albert, Measuring the User Experience, 2008

[S,R&P] Sharp, Rogers, and Preece, Interaction Design, 2002

[JN] Jakob Nielsen, Usability Engineering, 1994

[JR] Jeffrey Rubin, Handbook of Usability Testing, 1994

[JJG] Jesse James Garret, The Elements of User Experience, 2002