

Mensch-Maschine-Interaktion für den Master Elektrotechnik

Klausur vom 27. Januar 2015: Beispiellösungen

Jörn Loviscach

Versionsstand: 18. März 2016, 13:03



This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Germany License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/de/> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

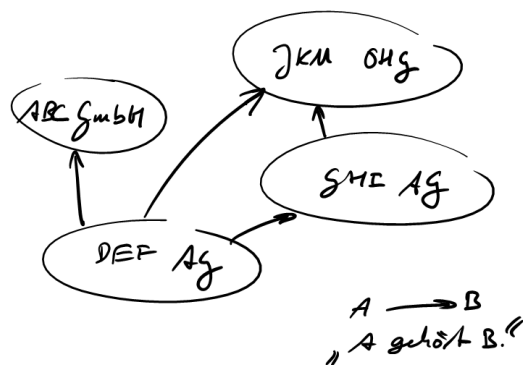
Anmerkungen zu alternativen Lösungen und zu möglichen Missverständnissen sind kursiv gesetzt.

1. Im Spiralmodell werden Entwicklungsphasen (insbesondere Implementation und Test) mehrfach durchlaufen, beim Wasserfallmodell nur einmal. Sehr typisch für die Anwendung eines Spiralmodells sind Methoden, um im Rahmen der ersten Spiralumläufe Prototypen zu erstellen, zum Beispiel diese drei Methoden: Paper Prototyping bedeutet, geplante Bildschirmhalte und deren Veränderung auf Papier zu simulieren. Wireframing bedeutet, geplante Bildschirmhalte skizzenhaft am Rechner zu zeigen, vielleicht mit einigen Interaktionsmöglichkeiten ausgestattet. „Wizard of Oz“ ist ein Verfahren, bei dem ein lauffähiges Programm auf einem Computer simuliert wird, indem ein Mensch an einem anderen, damit vernetzten Rechner die Aktionen manuell fernsteuert.

„Test“ usw. sind keine Methoden, sondern Phasen = Schritte.

2. Die Heatmap zeigt zweidimensionale Verteilungen; die Treemap zeigt relative Größen und ggf. Bestandteile und deren Größen, aber Unternehmen haben ja oft mehrere andere Unternehmen als Eigner; die Parallel Coordinates zeigen mehrdimensionale Verteilungen. Am besten passt hier ein Graph, weil er Verbindungen zwischen Objekten zeigt: An welchen anderen Unternehmen hält ein gegebenes Unternehmen Anteile? Wer wem gehört, kann man durch einen gerichteten Graph (Verbindungs Pfeile statt Verbindungslinien) klarstellen.

Den Umfang der Anteile – prozentual oder in Geld – könnte man noch durch die Stärke der Verbindung zeigen.



Den hier gemeinten Graph (Netz aus Knoten und Verbindungen) nicht mit dem Graph einer Funktion $y = f(x)$ verwechseln.

3. RETURN bzw. ESCAPE lösen die Standardfunktion eines Dialogs aus bzw. brechen ihn ab. Mit Tastaturkürzeln kann man direkt bestimmte Funktionen aufrufen, zum Beispiel unter Windows mit STRG+X die Auswahl ausschneiden. Mit der Tabulatortaste kann man durch Dialoge gehen und mit der Leertaste die Elemente bedienen. Unter Windows kann man mit ALT + unterstrichener Buchstabe die Menüs aufklappen und Menüeinträge aufrufen.

Einige von vielen weiteren Möglichkeiten: Scrollen mit BILD AUF und BILD AB, Springen im Dokument mit POS1 und ENDE, Startmenü usw. aufrufen mit WINDOWS-Taste, Hilfe aufrufen mit F1, Kontextmenü aufrufen mit Menü-Taste, nach Druck auf ALT mit den Cursortasten durch Menüs gehen.

4. Möglichkeit 1: Man stellt die mit einem Eye-Tracker über längere Zeit (Minuten) von möglicherweise vielen Benutzer gewonnenen Daten als Heatmap mit dem Bildschirminhalt als Hintergrund dar. So kann man erkennen, welche Bereiche des Bildschirms insgesamt wie häufig angesehen wurden.

Möglichkeit 2: Man zeichnet für einen Benutzer und eine kürzere Zeit (Sekunden) die Fixationspunkte und deren Verbindungen auf. So kann man zum Beispiel die Sprünge des Auges (Sakkaden) erkennen.

5. Bei „Think Aloud“ bittet ein Interviewer einen Benutzer (empirisches Verfahren!), eine BediENAufgabe zu lösen und dabei seine Gedanken laut auszusprechen; beim „Cognitive Walkthrough“ geht dagegen ein Experte in kleinen gedanklichen Schritten eine Aufgabe allein (analytisches Verfahren!) durch. „Think Aloud“ ist aufwändiger, weil es oft nicht reicht, einen einzigen Benutzer zu beobachten, und weil das laut Gedachte schwer zu dokumentieren ist. Außerdem könnte man Probleme übersehen, die beim schematisierten Vorgehen mit „Cognitive Walkthrough“ entdeckt würden. Andererseits können beim „Think Aloud“ durch die realen Benutzer Probleme auffallen, die man beim schematisierten Vorgehen mit „Cognitive Walkthrough“ übersehen hätte.

Alternative Punkte: Wie immer bei Tests mit realen Benutzern stellt sich auch beim „Think Aloud“ die Frage, ob sich der Benutzer schon genügend an die Software gewöhnt hat, ob er auf natürliche Art reagiert, ob er ehrlich sagt, was er denkt, und ob er überhaupt sagen kann, warum er etwas (nicht) tut.

6. Die übliche Frage ist, ob diese Beobachtung (4:1) auch rein zufällig passiert sein kann. Nullhypothese: Jeder Benutzer sagt mit einer Wahrscheinlichkeit von 50 %, dass das alte Design besser war, und ebenso mit einer Wahrscheinlichkeit von 50 %, dass das neue Design besser ist. Kann diese Nullhypothese abgelehnt und damit die gegenteilige Hypothese („Es gibt einen Unterschied.“) angenommen werden? Man untersucht also: Wenn diese Nullhypothese zutreffen sollte, wie groß ist dann die Wahrscheinlichkeit (der p -Wert), ein Ergebnis wie 4:1 oder noch ungleicher zu sehen?

Das könnte man bei dieser Aufgabe sogar abzählen: eine Möglichkeit für 5:0, fünf Möglichkeiten für 4:1, fünf Möglichkeiten für 1:4 und eine Möglichkeit für 0:5, macht 12 von insgesamt $2 \cdot 2 \cdot 2 \cdot 2 = 32$ Möglichkeiten, also $p = 12/32$, also viel zu groß, um „Signifikanz“ zu haben.

Von der Fragestellung her drängt sich bei dieser Aufgabe ein einseitiger Test auf. Dann wäre die Nullhypothese: Jeder Benutzer sagt mit einer Wahrscheinlichkeit von 50 % oder mehr, dass das neue Design schlechter ist. Das ist als Lösung für diese Aufgabe ebenfalls korrekt. Im Regelfall arbeitet man aber mit zweiseitigen Tests (und entsprechenden Nullhypothesen), es sei denn, es gibt prinzipielle Gründe, durch die die Verteilung effektiv einseitig wird. Denn einseitige Tests sind gefährlich: Sie führen bei gleicher Datenlage schneller zur Signifikanz; außerdem fassen sie die Kategorien „kein Unterschied“ und „unerwartetes (schädliches?) Ergebnis“ zu einer einzigen Kategorie zusammen, was nicht nur beim Test von Medikamenten bedenklich ist.^{c1}

^{c1} text added by jl

Der t -Test bietet sich hier nicht direkt an: Wovon sollen Mittelwert und Standardabweichung gebildet werden? Man könnte die Antworten einfach in 0 und 1 beziffern, aber diese Zufallsgröße wäre offensichtlich nicht normalverteilt.

7. Wenn man ein Programm schließen will, ohne seine Arbeit gespeichert zu haben, erscheint ein Warndialog. Eine Buchung lässt sich nicht abschließen, ohne dass man Name, Adresse und Bankverbindung eingegeben hat. Um ein neues Passwort festzulegen, muss man es zweimal identisch eingeben. Der Geldautomat gibt erst dann das Geld aus, wenn man seine Karte herausgenommen hat. Ein wichtiger Schalter ist so deutlich sichtbar angebracht, dass man daran erinnert wird, ihn richtig einzustellen.

Und viele weitere Möglichkeiten.

8. Die JND ist hier kein fester Prozentsatz des Reizes (hier: Gewichtskraft), sondern wird bei steigendem Reiz (hier: Verdoppelung von 10 auf 20 kg) relativ und sogar absolut kleiner. Das Weber-Fechner-Gesetz ist deshalb hier nicht plausibel. Man kann Stevens' Potenzgesetz ansetzen, mit einem Exponenten größer als eins, weil die JND hier mit wachsendem Reiz sinkt, die Kurve „Empfindung als Funktion vom Reiz“ mit wachsendem Reiz also immer steiler wird.

Die Gewichtskraft von 10 kg ist offensichtlich so weit oberhalb der Wahrnehmbarkeitsschwelle, dass man für die Wahrnehmbarkeitsschwelle in dieser Situation Null setzen kann.

9. Modell: Jeder Testbenutzer findet jedes Problem mit einer bestimmten, festen Wahrscheinlichkeit, und zwar Problem für Problem unabhängig voneinander und unabhängig von anderen Testbenutzern. Von den Problemen, die die bisherigen Testbenutzer nicht gefunden haben, findet der nächste Testbenutzer den entsprechenden Anteil. Die Zahl der nach N Testbenutzern nicht gefundenen Probleme fällt damit exponentiell mit N ; nach Nielsens empirischen Studien ist das oft schnell genug, um bei $N = 5$ abzurechnen. Einschränkungen: Die Zeit, nach der ein Problem gefunden wird, kann je nach Anwendungsbereich verschieden sein; außerdem ist die Annahme der Unabhängigkeit gewagt, z. B. weil in einer wenig genutzten Funktion eine Kaskade von Problemen stecken kann.

Fünf Nutzer können auch nicht die Vielfalt der gesamten Nutzerschaft abbilden.

Nielsen hat die Aussage so relativiert, dass man lieber verschiedene Entwicklungsstufen mit jeweils fünf Nutzern testen sollte.

10. *Zum Beispiel so:* Jedes Mal, wenn ein Touch-Ereignis eintrifft, stellt man fest, ob genau fünf Finger aufgesetzt sind. In diesem Fall bestimmt man den Schwerpunkt (Mittelwert) der fünf Fingerpositionen und wie weit sie im Mittel davon entfernt sind (ähnlich der Standardabweichung). Nach Ende der Berührung stellt man fest, ob in der Geste fünf Finger vorgekommen sind. Falls ja, prüft man, ob sich die mittlere Entfernung vom Anfang bis zum Ende der Geste stark verringert hat oder ob sich der Schwerpunkt vom Anfang bis zum Ende der Geste weit nach links bewegt hat. Im ersten Fall führt man aus, was beim Zusammenbewegen der Finger passieren soll, im zweiten Fall, was beim Wischen passieren soll.

Um Wischen von Panning zu unterscheiden (was nicht gefragt war), könnte man obendrein die Zeit messen.

Alternativ zum Beispiel so mit Pseudocode:

globale Variablen:

```
esWarenFünfFinger = false
schwerpunkt1
abstand1
schwerpunkt2
abstand2
```

Reaktion auf TouchBegin und TouchUpdate:

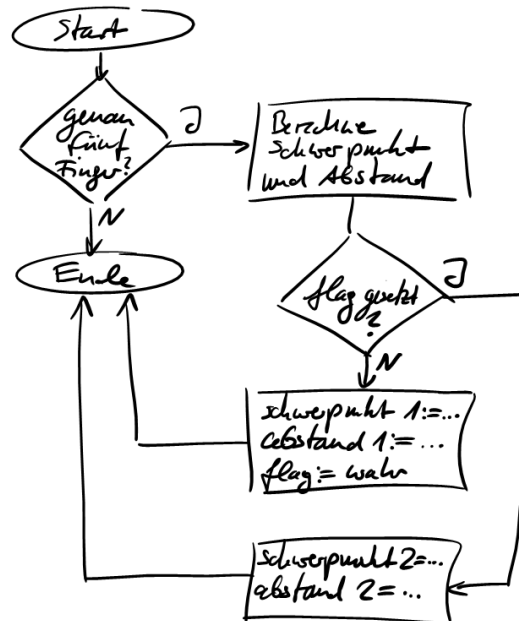
```
Wenn genau fünf Finger:
    berechne Schwerpunkt und Abstand
Wenn !esWarenFünfFinger:
    schwerpunkt1 und abstand1 schreiben
    esWarenFünfFinger = true
Sonst:
    schwerpunkt2 und abstand2 schreiben
```

Reaktion auf TouchEnd:

```
Wenn esWarenFünfFinger:  
  Wenn abstand1 - abstand2 > Schwelle:  
    behandle Zusammenbewegungsgeste  
  Wenn schwerpunkt1.x - schwerpunkt2.x > Schwelle:  
    behandle Wischgeste  
esWarenFünfFinger = false
```

Alternativ zum Beispiel so mit Flussdiagrammen:

Bei Touch Begin und Touch Update:



Bei Touch End:

