

Mensch-Maschine-Interaktion

Tutorium am 25. Juni

Übersicht

- Heuristische Evaluation
- Erläuterung der dritten Teilaufgabe
- Organisatorisches

Heuristische Evaluation

- Usability-Experten bewerten System/Prototyp anhand einfacher und allgemeiner Usability-Heuristiken
- Unabhängig von mehreren Experten durchzuführen
 - Daumenregel: 5 Experten finden 75% der Probleme
- Testen entweder lauffähiges System oder Prototypen
- Heuristiken / Kriterien:
 - Nielsen's 10 Heuristiken (1993; siehe Vorlesung 6)
 - Erweiterte Heuristiken ab 2001 (Nielsen, 2001)

Usability Heuristiken - Nielsen's 10 Regeln

- Erkennbarkeit des Systemstatus
- Übereinstimmung zwischen System und realer Welt
 - An die Sprache des Benutzers angepasst, den gängigen Konventionen entsprechend (Anordnung in natürlicher & logischer Reihenfolge)
- Benutzerkontrolle und -freiheit
 - Klar erkennbarer "Notausgang" zum Verlassen eines ungewollten Systemzustands (undo & redo)
- Konsistenz und Standards
 - Gleiche Situationen oder Aktionen sollten gleich benannt sein
- Fehlerverhütung
 - besser als eine gute Fehlerbeschreibung ist es, Fehler zu vermeiden

Usability Heuristiken - Nielsen's 10 Regeln

- Wiedererkennen statt sich erinnern
- Flexibilität und Effizienz der Benutzung
 - sollte für erfahrene sowie für unerfahrene Benutzer verwendbar sein, eigene Makros für häufig benutzte Funktionen ermöglichen
- Ästhetik und minimalistisches Design
 - Nur aktuell relevante Informationen anzeigen
- Hilfe beim Erkennen, Diagnostizieren und Beheben von Fehlern
 - Fehlermeldungen in allgemein verständlicher Sprache (kein Code), präzise Beschreibung des Problems mit Lösungsvorschlag
- Hilfe und Dokumentation

Heuristische Evaluation - Ablauf

- Trainingssitzung
 - Die Reviewer bekommen eine Einweisung in die heuristische Evaluation
- Evaluation
 - Jeder Reviewer evaluiert das Interface nach einer für alle, gleichen Liste an durchzuführenden Aktionen
 - Es sollte nicht nur aufgabenbasiert evaluiert werden, auch der gesamte Programmablauf sollte betrachtet werden
 - Ein Beobachter macht Notizen zu den gefundenen Fehlern
 - Die Reviewer dürfen sich erst nach der Evaluation austauschen

Heuristische Evaluation

- Ergebnisse
 - Probleme auflisten (gegen welche Prinzipien wurde verstoßen und warum)
 - Probleme sollen detailliert beschrieben werden
- Problem Bemessung
 - Wie gravierend und unvermeidbar ist das Problem?
 - Die Probleme von der Schwere nach von 1 bis 5 bewertet:
 - 0 - kein wirkliches Usability-Problem
 - 1 - oberflächliches Problem
 - 2 - geringes Usability-Problem
 - 3 - großes Usability-Problem - muss behoben werden
 - 4 - Usability Katastrophe - dringend beheben
 - Zum Ende werden alle identifizierten Problem gerankt

Heuristische Evaluation

- Beispiel:

Im Interface wird in einem Dialog „speichern“ zum Abspeichern angezeigt, in einem anderen Dialog wird diese Funktion dann „schreiben“ genannt. Diese wechselnde Terminologie könnte verwirren.

➔ mangelnde Konsistenz - Auf der Skala mit 3 bewertet

- Vorteil:

- Fehler kann günstig und schnell behoben werden

- Probleme:

- Experten sind keine echten Benutzer

- auch Heuristiken decken nicht alle Probleme ab

Teilaufgabe 3: Evaluation

- Szenario:

Ihr kommt nach Hause und möchtet via Device...

- das Licht einschalten
- die Wohnung durchlüften
- neue Nachrichten abfragen (Anrufbeantworter & Briefkasten)
- Badewasser einlassen
- etwas zu essen vorbereiten (Ofen vorheizen)
- Fenster schließen
- Heizung einschalten
- Ofen ausschalten
- das Licht ausschalten, um schlafen zu gehen

Evaluation - Vorgehensweise

- Teil 1: Einzelarbeit
 - Jedes Teammitglied startet das Programm der zu evaluierenden Gruppe zu jeweils 3 unabhängigen Testläufen (längere Pausen zwischen den Durchläufen)
 - Jeder Durchlauf wird auf jeweils anderem Server gestartet
 - Jedes Teammitglied erstellt in Einzelarbeit eine Mängelliste
- Teil 2: Gruppenarbeit
 - Allein erstellte Mängelliste mit Teampartner(n) abgleichen
 - Auftretende Probleme festhalten & in die fünf Kategorien einteilen
 - Positive Aspekte festhalten

Präsentation der Ergebnisse

- Jede Gruppe erstellt 2 Foliensätze:
 - Präsentation der eigenen GUI (5 min pro Vortrag/Gruppe)
 - Designentscheidungen erklären
 - Infos zur Implementation
 - Ggf. einen Interaktionsablauf mit Screenshots darstellen
 - Präsentation der Evaluation der fremden GUI (5 min)
 - Jeweils einen Punkt aus jeder der fünf Kategorien erläutern
 - Positive Aspekte aufzeigen
- Folien bitte frühzeitig an uns schicken!!!!!!

Wer evaluiert wen?

- Dosch, Deutenberg -> Schomann, Pohle, Najafi Tonekaboni
- Schomann, Pohle, Najafi Tonekaboni -> Bellmann, Lehmke
- Bellmann, Lehmke -> Michaelis, Heppner
- Michaelis, Heppner -> Hilsendeger, Walter
- Hilsendeger, Walter -> Möller, Heitländer
- Möller, Heitländer -> Dosch, Deutenberg

Testserver

- Server muss mit 1 oder 2 oder 3 also Übergabeparameter gestartet werden
 - `java -jar server.jar {1 | 2 | 3}`
- die aktuelle Version des Servers benötigt die Java 1.6
- bei Problemen oder Fragen zum Server:

Julia Tolksdorf M4-122,
jtolksdo@techfak

Organisatorisches

- Vorschlag:

Letzte zwei Tutorien (16. & 23. Juli) zu einem zusammenlegen?

- Beginn um 16:00 Uhr s.t.
- ca. 90 Minuten für 6 Gruppen (6 x 15 min)