

Praktikum Entwicklung von Mediensystemen mit iOS

WS 2011

Prof. Dr. Michael Rohs
michael.rohs@ifi.lmu.de
MHCI Lab, LMU München

Today

- Papier-Prototyp
- User-Test mit Papier-Prototyp
- Erfahrungen mit User-Test
- Start Software Prototyp

- Heuristische Evaluation

Timeline

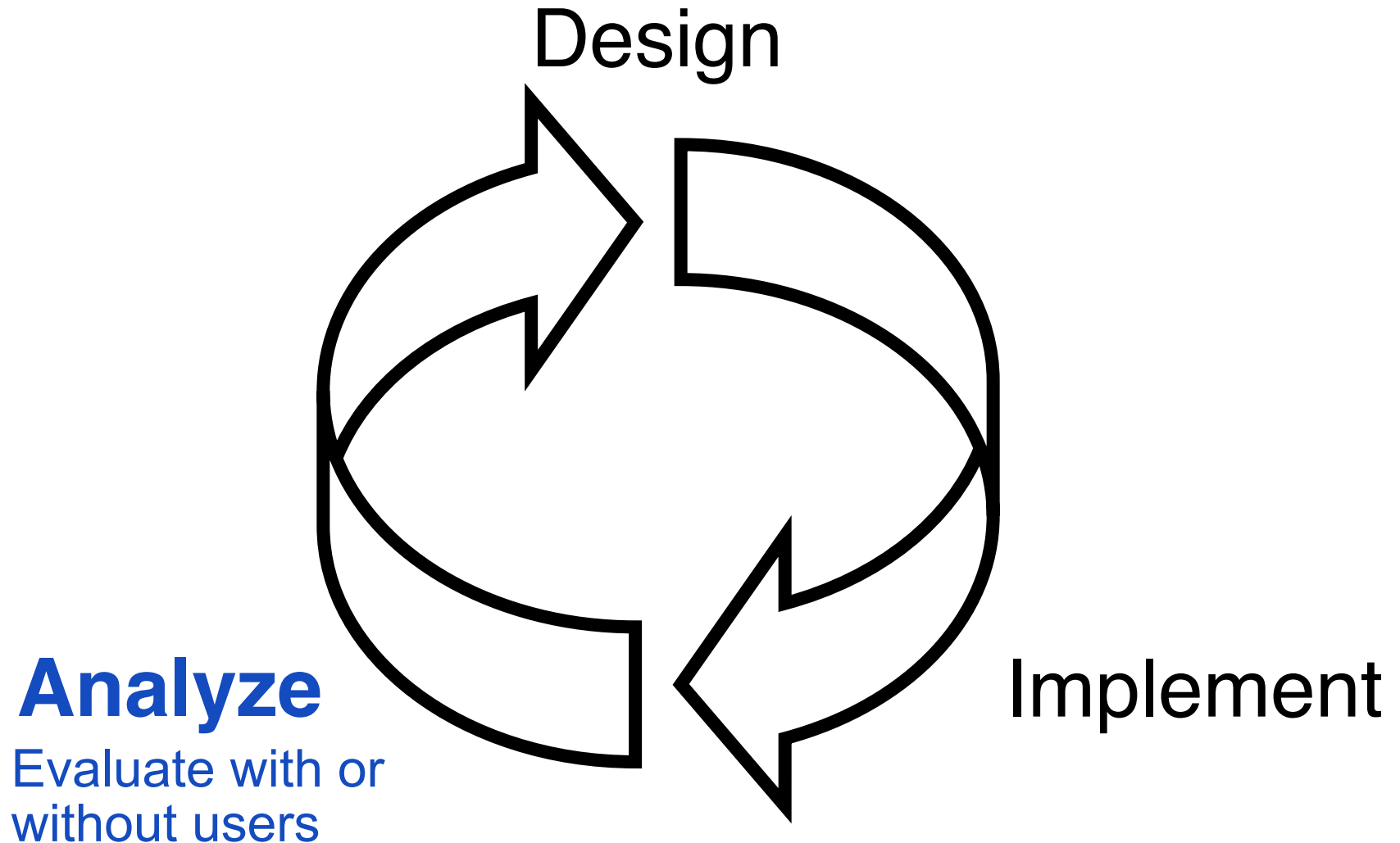
#	Date	Topic
1	19.10.2011	Introduction and overview of iOS
2	26.10.2011	App architecture, touch input, saving data
3	2.11.2011	Location, networking, sensors
4	9.11.2011	iOS 5, storyboards, automatic reference counting
5	16.11.2011	Interviews, storyboarding; brainstorming
6	30.11.2011	Paper prototyping test, start of software prototype
7	14.12.2011	Heuristic evaluation of software prototype
8	11.1.2012	Think-aloud user study
9	25.1.2012	Completion of software prototype
10	1.2.2012	Final presentation

vorige Aufgabe: Papier- & Software-Prototyp

- Erstellen Sie einen Papier-Prototyp, der die einzelnen Screens/UI-Zustände und Übergänge dazwischen darstellt.
- Führen Sie einen Benutzer-Test mit dem Papier-Prototyp mit 3 Benutzern (nicht aus dem PEM-Praktikum) durch
 - Typische Aufgabe festlegen
 - Benutzer bitten, diese Aufgabe mit dem Papier-Prototyp durchzuführen
 - Protokollieren, Probleme finden
 - Papier-Prototyp verbessern
- Schreiben Sie die Erfahrungen beim Test auf
 - Was ließ sich gut testen, was war problematisch
- Starten Sie den Software Prototyp

EVALUATION

DIA Cycle: When to evaluate?



Heuristic Evaluation

- Choose usability heuristics
 - (general usability principles, e.g., Nielsen's 10 Usability Principles)
- Step through tasks and check whether guidelines are followed
- Severity rating for each problem (Nielsen)
 - 0 = I don't agree this is a problem at all
 - 1 = cosmetic problem
 - 2 = minor usability problem, low priority to fix
 - 3 = major usability problem, high priority to fix
 - 4 = usability catastrophe, imperative to fix before release
- + Quick and cheap
- Subjective (have several independent evaluators)
 - See also: www.useit.com/papers/heuristic

10 Usability Heuristics (Jakob Nielsen)

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Help users recognize, diagnose, and recover from errors
10. Help and documentation



http://www.useit.com/papers/heuristic/heuristic_list.html

Four Fundamental Concepts (Donald Norman)

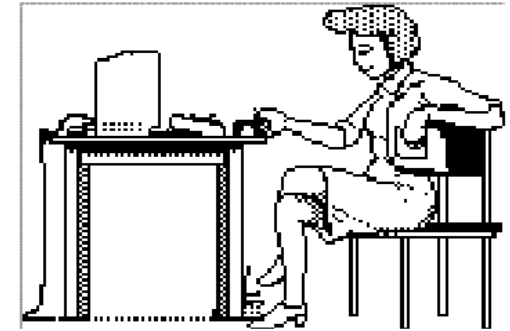


- Affordances & visibility
 - Affordances
 - http://www.jnd.org/dn.mss/affordances_and_design.html
 - Can the user tell the state of the system and the alternatives for action by looking at the system?
- Conceptual models
 - Is the user able to predict how actions affect the system?
- Natural mapping
 - Is it possible to determine the relationships between actions and results, between controls and effects?
- Feedback
 - Does the user receive full and continuous feedback about the results of actions?

User Interface Guidelines

- Concrete guidelines for look-and-feel and behavior
 - Visual appearance, e.g., icon design
 - Purpose of user interface elements
 - Layout of user interface elements
 - Behavior, conventions of system features
- Android User Interface Guidelines
 - http://developer.android.com/guide/practices/ui_guidelines/index.html
- iOS Human Interface Guidelines
 - <http://developer.apple.com/library/ios/documentation/userexperience/conceptual/mobilehig/MobileHIG.pdf>
 - Aesthetic integrity, consistency, direct manipulation, feedback, metaphors, user control, ...

Silent Observation



Source: Saul Greenberg

- Designer watches user in lab or in natural environment while working on one of the tasks
- No communication during observation
- + Helps discover big problems
- No understanding of decision process (that may be wrong) or user's mental model, opinions, or feelings

Think Aloud



Source: Saul Greenberg

- As Silent Observation, but user is asked to say aloud
 - What he thinks is happening (state)
 - What he is trying to achieve (goals)
 - Why he is doing something specific (actions)
- Most common method in industry
- + Good to get some insight into user's thinking, but:
 - Talking is hard while focusing on a task
 - Feels weird for most users to talk aloud
 - Conscious talking can change behavior

Evaluation in the Mobile Context

- Context of use needs to be taken into account
 - Factors: User, activity, device, environment
- Usage “on the move”
 - Physically moving: walking, driving a car, traveling as a passenger
 - Being in different places: away from office environment or home
- Difficult to collect data in the field
 - Recording interaction
 - Capturing context
 - Controlling experimental conditions



AUFGABE

Aufgabe

- Ziel der Aufgabe: Test des Software-Prototyps ohne Benutzer mittels heuristischer Evaluation
- Heuristiken sind allgemeine Regeln, die beim Entwurf der Benutzungsschnittstelle beachtet werden sollten.
 - Verwenden Sie Niensens aktualisierte Liste von “Ten Usability Heuristics” als Basis (Folien, sowie www.useit.com/papers/heuristic/heuristic_list.html).
 - Fügen Sie dieser Liste basierend auf der bisherigen Konzeptentwicklung mindestens eine weitere Regel hinzu, die spezifisch für Ihr Projekt sind.

Aufgabe

- Vorgehen
 - Einigen Sie sich im Team auf Szenario und Test-Aufgabe
 - Jedes Team-Mitglied führt unabhängig (!) eine heuristische Evaluation durch
 - Bewerten Sie die Wichtigkeit der gefundenen Usability-Probleme
 - Tragen Sie die Resultate zusammen, vergleichen Sie die Resultate und ordnen Sie nach Wichtigkeit
 - Ändern Sie den Software-Prototyp, um die Probleme zu lösen

Aufgabe

- Lesen Sie die folgenden Quellen als Hintergrundinformation:
 - Heuristic evaluation:
www.useit.com/papers/heuristic
 - How to conduct HE:
www.useit.com/papers/heuristic/heuristic_evaluation.html
 - On severity ratings:
www.useit.com/papers/heuristic/severityrating.html

Aufgabe

- Beim nächsten Treffen soll jede Gruppe folgendes vortragen:
 - Liste zusätzlicher Heuristiken (2-4)
 - Gefundene Usability-Probleme (Text / Bilder / mündlicher Vortrag)
 - Bewertung der Wichtigkeit (Severity Rating) jedes Problems
 - Beschreibung der Änderungen am Software Prototyp (Text / Bilder / mündlicher Vortrag)
 - Kurze mündliche Beschreibung der Erfahrungen mit heuristischer Evaluation (Bewertung, Aufwand, etc.)
- Arbeiten Sie weiter an Ihrem Software-Prototyp. Zeigen Sie beim nächsten Treffen kurz den aktuellen Stand.