

Entwerfen und Verwerfen: UI-Konzeption für FNT Command

Fabian Hermann
Fraunhofer IAO
Nobelstr. 12
70569 Stuttgart
fabian.hermann@iao.fraunhofer.de
www.hci.iao.fraunhofer.de

Detlef Klugseder
FNT-GmbH
Roehlinger Strasse 11
73479 Ellwangen
d.klugseder@fnt.de
Internet www.fnt.de

Abstract

Anhand eines Projekts zur UI-Konzeption der FNT GmbH werden typische Schwierigkeiten bei der Konzeption von komplexer Benutzungsschnittstellen diskutiert, und wie man in der Praxis dennoch zu guten Ergebnissen kommt:

Nutzerrecherche auch mit begrenzten Ressourcen, Gestaltungsentscheidungen auch auf unvollständiger Informationsbasis, systematische Konzeption und Proof of Concept mit strukturellen Prototypen.

Keywords

UI-Konzeption, Prototyping, Anforderungsanalyse, benutzerzentrierte Gestaltung

1.0 Einleitung

Das Konzipieren von UIs bedeutet Informationen und Anforderungen zu sammeln und abzuwägen, Ideen zu finden, zu entwickeln, und sie oft auch wieder zu verwerfen, Entwürfe klar durchzustrukturieren und (manchmal) etwas hinbiegen, Details zusammenpuzzeln und dabei das große Ganze nicht aus den Augen verlieren. Oft ist dies weit entfernt von einem systematischen Engineering, aber trotzdem gibt es einiges an Vorgehensweisen, die zu beherzigen sich lohnen. Einige Aspekte diskutieren wir anhand eines Projekts, das die FNT-GmbH gemeinsam mit dem Fraunhofer IAO durchgeführt hat. Dabei wurde ein neues UI-Konzept für „Command“ entwickelt, der FNT-Produktfamilie für das IT Service Management auf der Basis von ITIL (Information Technology Infrastructure Library, vgl. <http://www.itil.co.uk/>). Ziel des Projekts war es vor allem, eine tragfähige Sichten- und Navigationsstruktur zu erarbeiten, die in einer nächsten Version des UIs umgesetzt werden kann.

2.0 Der Ausgangspunkt: Komplexe Datenstrukturen, komplexe Interaktion

Die Motivation, das aktuell verwendete UI zu überarbeiten, ergab sich aus

einer Reihe bekannter Probleme, von denen einige in der Gesamtstruktur des GUIs begründet liegen. Diese Probleme entstanden, wie so oft, mit der historischen Entwicklung des Produkts: Mit der Zeit wuchs nicht nur der Funktionalitätsumfang, sondern auch die Komplexität der mit dem System verwalteten Daten- und damit auch Interaktionsstrukturen.

Ein typisches Beispiel für eine komplexe Interaktion ist, dass der Benutzer ein Objekt neu anlegt (z.B. einen „Dienst“), zu dem er ein anderes Objekt zuordnen muss (z.B. einen „Provider“). Wenn dieses Objekt dann ebenfalls neu angelegt werden muss, so sind bereits zwei, hierarchisch aufeinander bezogene Aktionen „neu anlegen“ notwendig. In der Praxis werden bis zu fünf miteinander verbundene Objekte neu angelegt.

Es war eine der harten Nüsse für die Neukonzeption, diese Komplexität im User Interface abzubilden, navigierbar und verständlich zu machen. Die aktuell verwendete Gestaltungslösung ermöglicht zwar einen sehr flexiblen Umgang, zeigte die hierarchische Struktur der Interaktion aber nicht an. Auch wurde der Benutzer kaum geführt, sodass die Struktur schwer verständlich und erlernbar war.

2.1 Die Informationslage zu Beginn

Dieses und weitere Probleme waren im Wesentlichen aus Kundenkontakten bekannt. Nicht nur der Vertrieb, sondern auch Projektleiter und die Entwicklung stehen häufig in sehr intensivem Kontakt mit Kunden und Benutzern, v.a. wenn Funktionen, Datenstrukturen und UI-Lösungen kundenspezifisch angepasst werden, was sehr häufig geschieht. In der Regel finden dafür Anforderungsworkshops mit den Kunden statt, in denen v.a. die Funktionalität, dabei aber auch Schwierigkeiten mit dem UI oder Anforderungen diskutiert werden. Dies liefert einiges an Erkenntnissen, wo die Probleme liegen. Weitergehende Informationen z.B. über Nutzerverhalten wurden aber nicht systematisch, z.B. durch Benutzertests oder Aufgabenanalysen erfasst.

2.2 The Design Gap

Die Lücke zwischen Anforderungen und Gestaltung besteht selten darin, dass man vor dem leeren Blatt sitzt und ratlos ist. Vielmehr liegen oft genügend Ideen auf dem Tisch, wie man es machen könnte: oft aus Erfahrungen der Teammitglieder, oder z.B. auch systematischen Best-Practice-Recherchen.

Für das Beispiel „neu Anlegen“ lag schon sehr früh der Vorschlag auf dem

Tisch, einen Wizard zu gestalten, um eine stärkere Benutzerführung zu erreichen. Für das Anlegen mehrerer, hierarchisch verknüpfter Objekte wurden zwei Ideen diskutiert:

- zum einen die Darstellung eines Unter-Objekts in jeweils einem neuen Fenster, was bei den möglichen fünf Hierarchieebenen zu einem Stapel von Fenstern führt
- zum anderen die Darstellung aller Schritte für alle Objekte in einem Fenster, was Entscheidungen des Nutzers im Wizard notwendig macht

Das Problem besteht also meistens nicht im Mangel an Ideen, sondern in deren Bewertung. Nur selten sind Ideen offensichtlich als Verbesserungen erkennbar, und wenn, dann betrifft dies kleine, isolierte Probleme. Wenn es aber um die großen Dinge geht – Sichten-, und Dialogstruktur, Templates usw. – ist die Bewertung der Ideen entscheidend.

2.3 Informationslöcher

Diese Bewertung leidet oft unter fehlenden Informationen: Man diskutiert Vor- und Nachteile verschiedener Vorschläge, aber die Entscheidung lässt sich nicht treffen, weil Detail-Informationen über die Arbeitsweisen der Benutzer fehlen. Z.B. fehlt häufig eine genaue Einschätzung, wie häufig bestimmte Interaktionssequenzen ausgeführt werden und dementsprechend, wie wichtig ihre Unterstützung ist.

Für das Beispiel „neu Anlegen“ mehrerer, hierarchisch verknüpfter Datenobjekten wäre es z.B. hilfreich gewesen zu wissen, wie oft tatsächlich solche verknüpfte Objekte gemeinsam angelegt werden und wie tief die Interaktionshierarchie typischerweise werden kann.

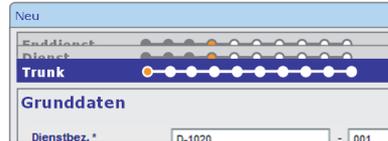


Abbildung 1: Visueller Prototyp für eine Visualisierung hierarchischer Interaktionssequenzen

Solche Informationslöcher können z.B. durch Benutzertests gestopft werden. Kurzfristige Tests mit geringem Aufwand sind aber schwer zu realisieren. Da solche „kleinen“ Entscheidungen ständig getroffen werden müssen, lohnt sich der Aufwand eines Tests oft nicht bzw. ist unter dem üblichen Zeitdruck nicht realisierbar.

Pragmatisch kann man die Entscheidung dann nur trotz der geringen Information zu fällen. Wichtig dabei ist:

- Sich bewusst machen, dass die Entscheidung uninformatiert getroffen werden muss; viele Team-Diskussionen drehen sich im Kreis, weil man Für und Wider zu lange abwägt, statt die Entscheidung auf der Basis der vorhandenen Informationen zu wagen
- Annahmen treffen, was vermutlich die beste Lösung ist, und die Design-Entscheidung in einem speziell dafür gedachten Protokoll festhalten. Das Protokoll sollte die Entscheidung und die Gründe und hinter stehenden Annahmen festhalten, um zu verhindern, dass

dieselben Diskussionen mehrfach geführt werden.

- Nicht von Extrem-Situationen ausgehen: Wenn genaue Kenntnisse über die Benutzer fehlen, werden oft Extrem-Fälle zugrunde gelegt: „Die Liste kann bis zu 500 Einträge haben“, auch wenn 95 % der Nutzer in 95 % der Fällen mit 80 Einträgen auskommen
- Ersatz-User fragen: interne Leute fragen, die Nutzer-Kenntnisse haben (Vertrieb, Schulungspersonal, fachliche Experten usw.) Auch wenn diese Personen keine „echten“ Benutzer sind und deshalb für den Usability-Test nicht in Frage kommen, können sie doch gute Angaben über typische Verwendungsweisen machen und sind für die Konzeption sehr hilfreich.

2.4 Ideen ausprobieren: Prototyping

Das große Potential des schnellen Prototypings, v.a. mit einfachen Mitteln wie Papier oder Powerpoint, liegt im schnellen, informellen Ausprobieren von Ideen: Sehr schnell lässt sich mit Papier und Bleistift darstellen, wie eine Grundidee aussieht und welche Konsequenzen ihre Umsetzung hat.

Von den beiden Ansätzen für das Beispiel „Neu Anlegen“ verfolgten wir zuerst den Wizard, der ermöglichen sollte, mehrere, hierarchisch verknüpfte Objekte in einem Fenster anzulegen. Dabei lag unser Fokus zunächst darauf, eine Visualisierung zu entwerfen, die dem

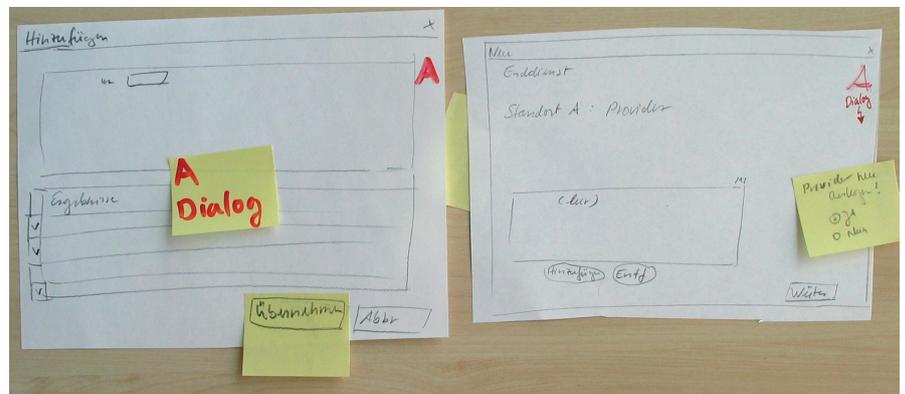


Abbildung 2: Papier-Prototyp. Sichten als Papierblätter, Bedienelemente als Bleistiftskizzen und Klebezettel, die leicht zwischen und auf Sichten verschoben und arrangiert werden

Benutzer diese komplexe Interaktionsstruktur verdeutlichen sollte (vgl. Abb. 2). Diese, für den Ansatz entscheidende Darstellung ließ sich tatsächlich in mehreren Schritten gut lösen, was sich auch in ersten, allerdings informellen Umfragen belegen ließ. Parallel aber zeigten sich bei der Arbeit mit Papier-Prototypen (vgl. Abb. 1) Schwierigkeiten mit diesem Ansatz, die wir nicht vorausgesehen hatten, v.a. wurde durch die hierarchische Struktur innerhalb eines Fensters, mit dem nacheinander mehrere Objekte angelegt und gespeichert werden, die Dialognavigation des Wizards (Zurück, Weiter, Fertig stellen) uneindeutig. Diese Schwierigkeiten, die sich im Papier-Prototyp zeigten, erschienen uns schließlich so kritisch, dass wir den Ansatz doch komplett verwarfen und auf das – unelegantere – Konzept mehrerer Wizards in jeweils eigenen, modalen Dialogen umsetzten.

Dieses Beispiel zeigt eine wesentliche Stärke des Papier-Prototypen: dass sich schnell prüfen lässt, ob sich ein Ansatz klar und konsistent durchziehen lässt oder, ob aus einer großen Lösung im Detail tausend kleine Probleme folgen. Dies ist ein Kernpunkt für die Entwicklung guter, ergonomischer UI-Konzepte: Alternativen schnell entwerfen, durchdenken und auf Konsistenz prüfen.

Das Beispiel zeigt auch einmal mehr, dass gut benutzbare Konzepte auch die nahe liegenden, „langweiligen“ sein können: Statt eines neuartigen, potentiell vielleicht eleganten „hierarchischen Wizards“ ein einfacher, in seiner Funktion wohl definierter und dem Benutzer bekannter Stapel modaler Fenster.

3.0 Fazit

Zwei für die UI-Konzeption wesentliche Erfahrungen liegen uns am Herzen:

- Optimal informierte Design-Entscheidungen aufgrund von Nutzerdaten sind (leider) nicht immer möglich, umso wichtiger ist ein strukturierter Umgang mit ihnen: zugrunde liegende Annahmen klären und dokumentieren.
- Für ergonomische Ergebnisse muss die UI-Struktur schon in der Konzeptionsphase „zu Ende gedacht“ werden, sodass die UI-Struktur „passt“ und man sich nicht versteckte, strukturelle Probleme einhandelt, die Verständnisproblemen beim Benutzer führen. Das schnelle Prototyping mit Papier ist dafür der richtige Weg, weil strukturelle Aspekte betont und für das konzeptuelle Stadium überkonkrete Ausarbeitung verhindert wird.

4.0 Referenzen

Information Technology Infrastructure Library, vgl. <http://www.itil.co.uk/>

»Es ist erlaubt digitale und Kopien in Papierform des ganzen Papers oder Teilen davon für den persönlichen Gebrauch oder zur Verwendung in Lehrveranstaltungen zu erstellen. Der Verkauf oder gewerbliche Vertrieb ist untersagt. Rückfragen sind zu stellen an den Vorstand des GC UPA e.V. (Postfach 80 06 46, 70506 Stuttgart). Proceedings of the 4th annual GC UPA Track Gelsenkirchen, September 2006 © 2006 German Chapter of the UPA e.V.«

