

31.8.–3.9.2015
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence

Wie geht das zusammen?

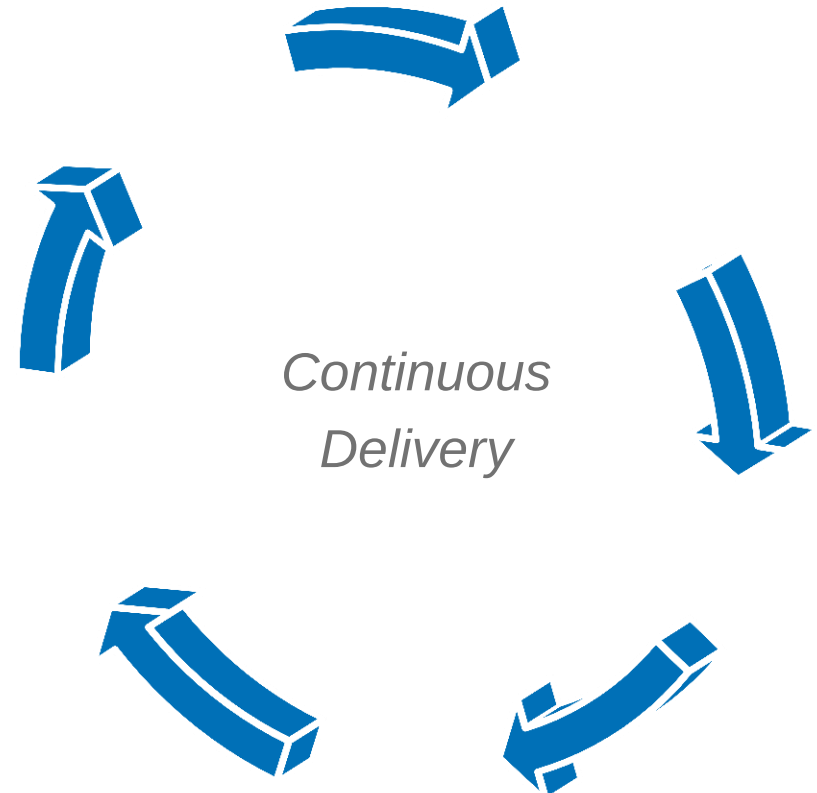
GUI-Redesign in einer Continuous Delivery-Umgebung

Martin Aschoff,

AGNITAS AG

Continuous Delivery

- ▶ ständige Release-Bereitschaft
- ▶ Entwicklung in kleinen Schritten mit Hilfe bewährter Muster
- ▶ Testing und Deployment über automatisierte Tool-Chain



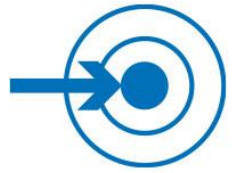
GUI Redesign

- ▶ Änderung der kompletten GUI in einem einzigen Schritt
(notwendig, um Inkonsistenzen zu vermeiden)



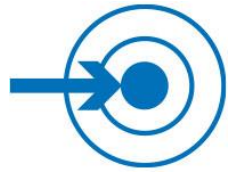
Ausgangslage

- ▶ Software als SaaS, Private Cloud und Lizenz
- ▶ über 1200 aktive Nutzer, teilweise 10-20 Stunden/Woche
- ▶ GUI war 5 Jahre alt
- ▶ GUI lief auch mit IE 6 einwandfrei
- ▶ wenig Javascript, kaum Ajax
- ▶ Verfallserscheinungen in CSS-Dateien



Redesign Ziele (1)

- ▶ responsive Layout
- ▶ verzerrungsfreie Skalierung
- ▶ Bedienung für Touchscreens geeignet
- ▶ Dynamische GUI durch verstärkten Ajax-Einsatz



Redesign Ziele (2)

- ▶ bessere Basis für Usability-Verbesserungen
- ▶ Persistierung von GUI Einstellungen im Browser Storage
- ▶ Flat Design (d.h. keine Rahmen, keine auffälligen Farbverläufe, keine Schatten, keine 3D-Effekte)
- ▶ Icon Font

Alte GUI


**EMM
2014**

agnitas_asp (Agnitas_ASP) | Abmelden | Hilfe

- » Dashboard
- » Zusammenfassung
 - » Kalender
- » Mailings
- » Kampagnen
- » Empfänger
- » Statistik
- » Zielgruppen
- » Mailing-Listen
- » Formulare & Aktionen
- » Datei hochladen
- » Verwaltung
- » Grid CMS
- » Medienpool

Zuletzt benutzte Mailings

Liste

Vorschau



31.01.2014
Kundenbefragung



30.01.2014
Kundenbefragung



27.01.2014
Kundenbefragung



23.01.2014 2-Wege-Authentifizierung



09.01.2014 Sexy Trippermails



16.01.2014 Marketing



31.12.2013



24.12.2013



31.12.2013



23.01.2014



09.01.2014



16.01.2014

Statistik

Mailing (einzel)

Zuletzt versendet

Anzeigen

Mailing Kennzahlen

Ausgewähltes Mailing: 31.01.2014 Kundenbefragung EN

Versendete Emails	44
Öffnungen	2 (4,55%)
Klickende Empfänger	0 (0,00%)
Abmeldungen	0 (0,00%)
Hardbounce	0 (0,00%)

Neue Mailings erstellen

Mailing

Schablone Keine Schablone

- » [Neues Mailing erstellen](#)
- » [Neues Mailing mit Assistent erstellen](#)
- » [Neue Schablone erstellen](#)

Empfänger verwalten

- » [Empfängerübersicht](#)
- » [Blacklist pflegen](#)
- » [Empfänger hinzufügen](#)
- » [Anredearten](#)
- » [Empfänger importieren](#)
- » [Empfänger exportieren](#)

Neue GUI



2015

Zusammenfassung

Dashboard

Zusammenfassung

Kalender

Mailings

Kampagnen

Empfänger

Statistik

Zielgruppen

Mailing-Listen

Formulare & Aktionen

Datei hochladen

Verwaltung

Grid CMS

Medienpool

Eingeloggt als Marketing AGNITAS

Abmelden

Altes Design

Zuletzt benutzte Mailings

Liste

Vorschau

+ Neu



INSIDE

EMM in neuem Design: Jetzt ausprobieren!



Newsletter Ideen

Termine und Anlässe für Mailings im März



Anmeldebestätigung

EMM-Schulung am 23.03.15 in München



Anmeldebestätigung

EMM-Schulung am 23.03.15 in München



EMM Schulung am 23. März in München

Kostenlose Live-Schulung im neuen EMM-Design



Datenaustausch leicht gemacht

Schnittstellen im Überblick



Automatisierte E-Mail-Kampagnen

A/B-Test mit Vorlage erstellen



EMM in a new design

Have a look and test it!

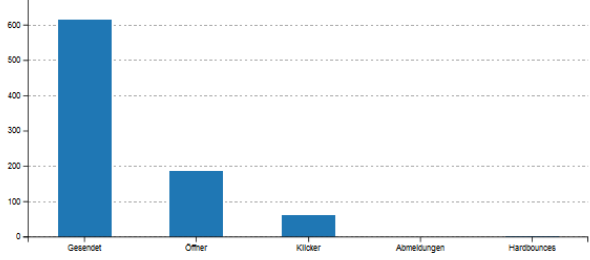


Anmeldebestätigung

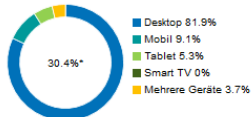
EMM-Schulung am 23.03.15 in München

Statistik

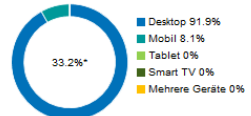
2015-01-26 Newsletter Part 1: EMM Redesign



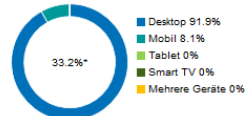
Öffner



* aller Empfänger



Klicker



* aller Öffner

Februar 2015

Monatsansicht

Wochenansicht

Anzeigen

<

Heute

>

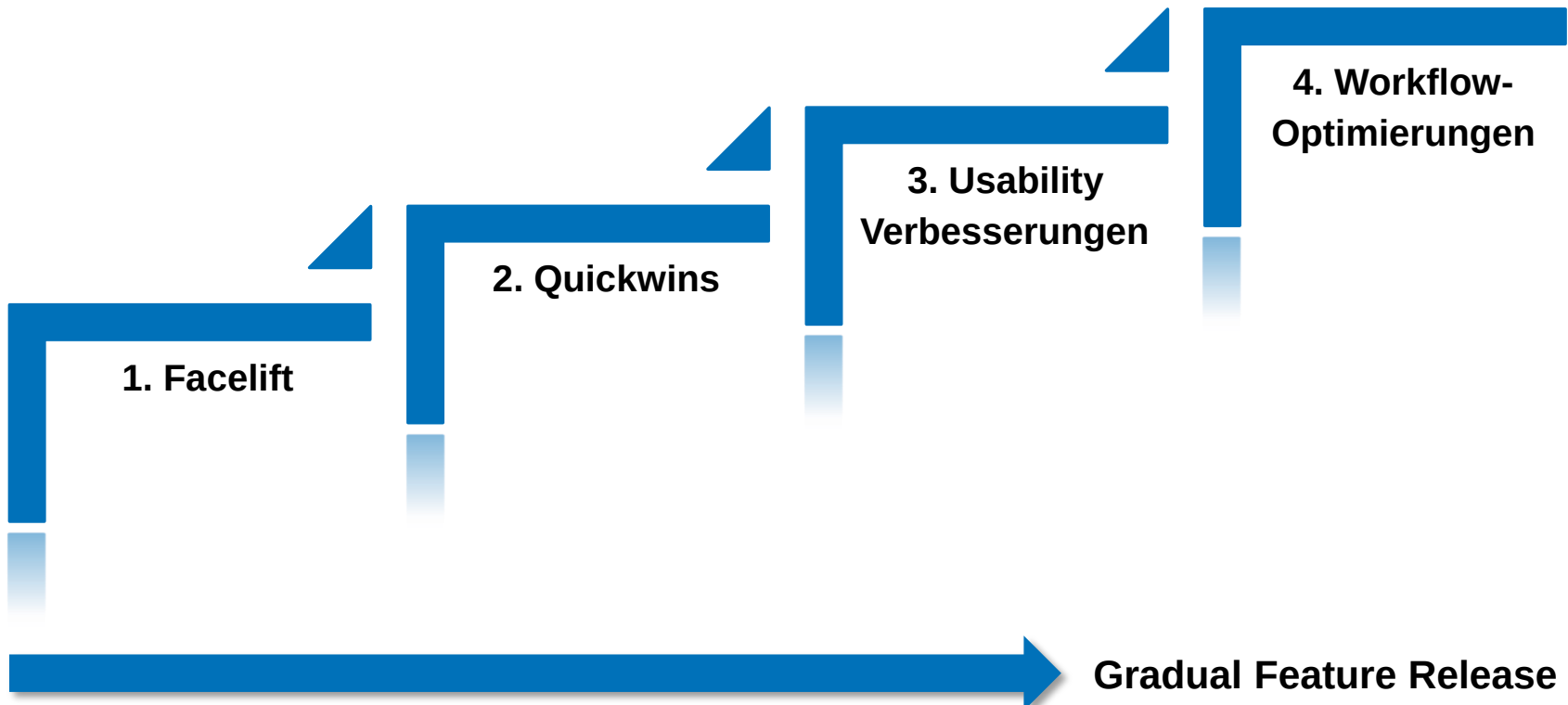
Februar

2015

KW	Montag	Dienstag	Mittwoch	Donnerstag	Freitag	Samstag	Sonntag	Ungeplante Mailings
5	26	27	28	29	30	31	1	Experiment! - 08.05.2014 Agiles E-Mail-Marketing Teil 1: Realtime Content xxx Agnitas Infoletter
6	2	Anmeldebestätigung Schu...	4	2015-02-03 Testphase EM	05.02.2015 Automatisierte	7	8	

Continuous Delivery-Muster umgesetzt für GUI Redesign (1)

Aufteilung des Redesigns in mehrere Stufen:



Continuous Delivery-Muster umgesetzt für GUI Redesign (2)

Definition des Mindest-Funktionsumfangs für das Facelift

→ *Minimum Viable Product (MVP)*

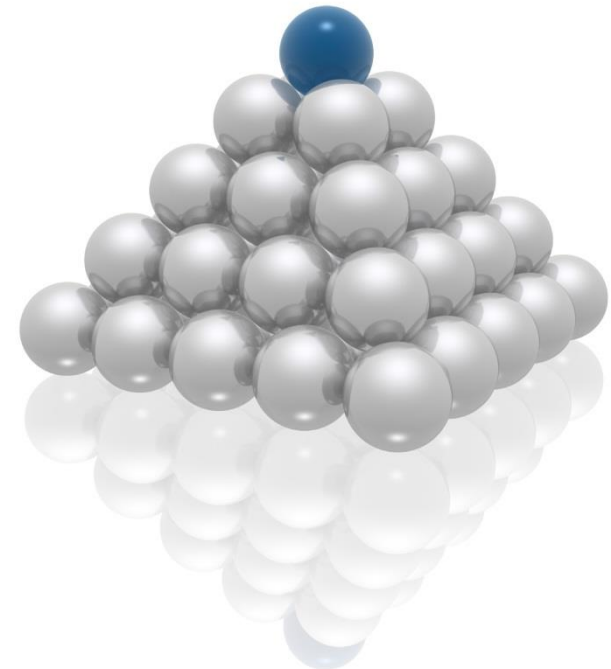


Continuous Delivery-Muster umgesetzt für GUI Redesign (3)

Aufteilung der Umsetzung des restlichen Funktionsumfangs

- ▶ selten genutzte Features
- ▶ Aufpreis-Features
- ▶ interne Features

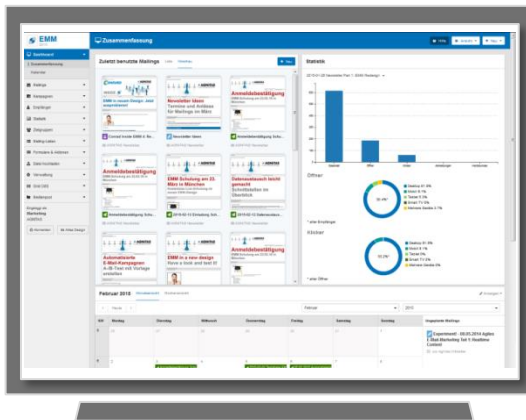
→ *Gradual Feature Release*



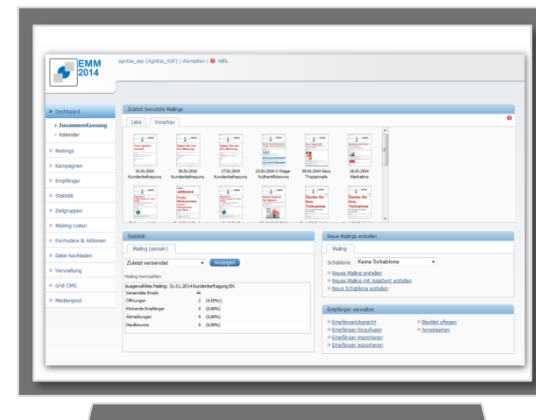
Continuous Delivery-Muster umgesetzt für GUI Redesign (4)

Parallelbetrieb der alten und der neuen GUI

→ *Rival Feature*



Neue GUI

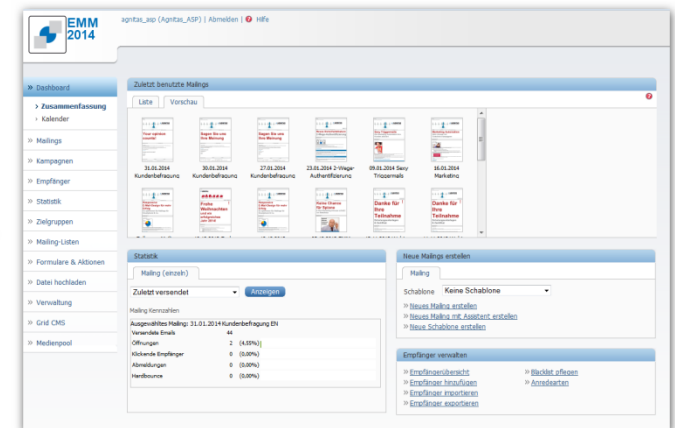
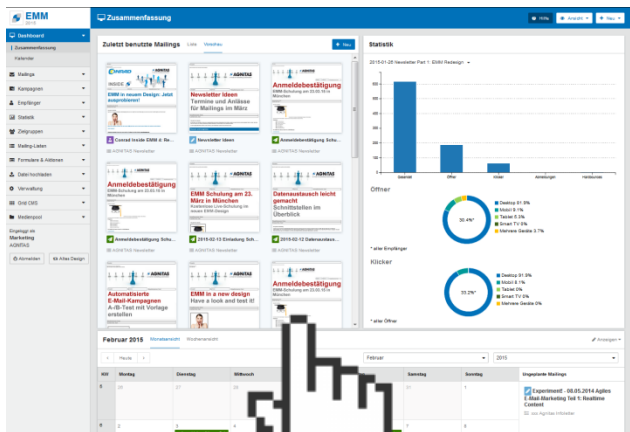


Alte GUI

Continuous Delivery-Muster umgesetzt für GUI Redesign (5)

GUI-Wechsel im laufenden Betrieb

→ *Feature Toggle*



Technik hinter der GUI-Umschaltung (1)

- ▶ anfänglich leerer Ordner für neuen View-Layer (/view), der sukzessive mit Dateien der neuen GUI aufgefüllt wird
- ▶ für GUI Redesign wird dieser Ordner über den Ordner des alten View Layers (/jsp) kopiert
- ▶ Deploy-Skript legt eine Applikation mit jsp-Ordner an
→ zur Nutzung durch Kunden
- ▶ Deploy-Skript legt zweite Applikation mit jsp-Ordner an, über den Inhalt des view-Ordners kopiert wird → für Testing und interne Nutzung

Technik hinter der GUI-Umschaltung (2)

- ▶ beide Applikationen laufen in Webcontainer (Tomcat), es muss aber nicht der gleiche sein
- ▶ Aufruf der jeweils anderen Applikation über extra Button, der diese per URL mit Zielseiten-Parametern sowie verschlüsseltem Token mit Authentifizierungs-Infos (Instanz, Mandant und Nutzer) und Timestamp aufruft
- ▶ andere Applikation entschlüsselt Token, prüft Nutzer, Mandant, Instanz und Timestamp, loggt User automatisch ein und ruft in der URL angegebene Seite auf
- ▶ Ver- und Entschlüsselung des Token basieren auf symmetrischem Verschlüsselungsverfahren + Salt in gemeinsamer DB

Best Practise für die Aufteilung des Redesigns

(genau so hätten wir es machen sollen)



1. Facelift

- ▶ Layout: Flächen, Formen, Farben, Positionen
- ▶ Bilder & Grafiken
- ▶ Icons & Fonts
- ▶ Effekte (keine Funktionslinien, keine runden Ecken, keine Schatten → Flat Design)

2. Quickwins

- ▶ Usability-Verbesserungen mit wenig Aufwand wie
 - Vereinheitlichung von Bezeichnungen
 - Position von Bedienelementen, um Mauswege zu verkürzen
 - Ändern der Spaltenbreiten
 - Reihenfolge von Formular-Boxen
 - Austausch von Widgets (Datepicker, Select2, etc.)

3. Usability-Verbesserungen

- ▶ geänderte Menü-Strukturen, um Klickwege zu verkürzen
- ▶ Ausbau des Dashboards, um Klicks zu sparen
- ▶ neuer Quellcode-Editor mit Syntax-Highlighting und Fullscreen-Modus
- ▶ WYSIWYG-Editor mit neuer Skin und Fullscreen-Modus
- ▶ neuer Zielgruppen-Builder

4. Workflow-Optimierungen

- ▶ Umgestaltung bestehender Features (Split, Fusion, etc.)
- ▶ neue Wizards
- ▶ neue Features



Argumente gegen Gradual Feature Releases

Ein Facelift allein bietet
keinen Mehrwert

Zumindest sieht die GUI
schon mal besser aus

Das kann man nicht
voneinander trennen

Warum nicht?

Wenn ich das gleich
mitmache, dann spare ich
Arbeitszeit

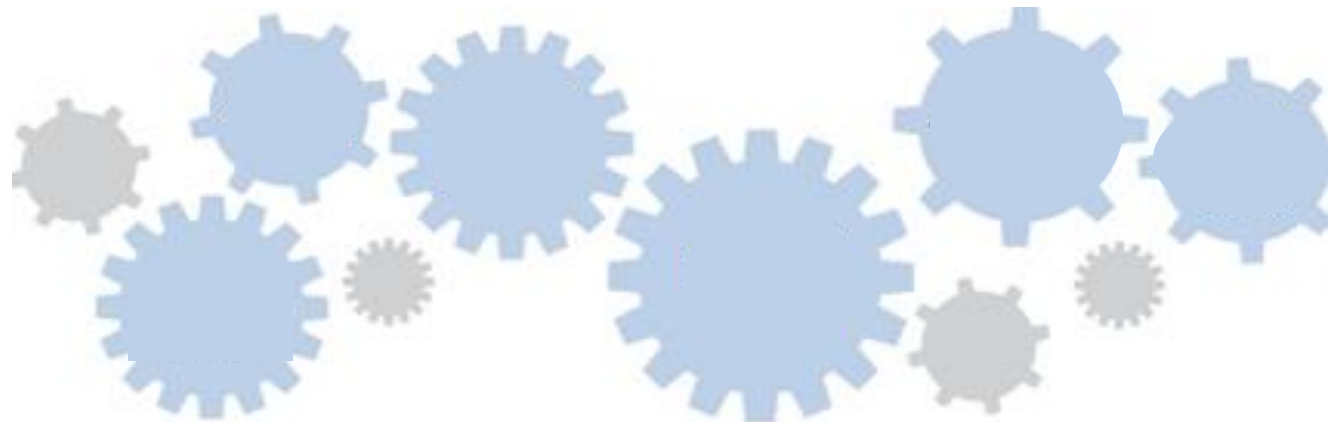
Eventuell, aber es gibt
schneller Feedback, das
einige Arbeit u.U. über-
flüssig macht

Das geht ganz schnell, ist
nur ein Quickwin

nur, wenn es nicht länger
als eine halbe Stunde
dauern wird

Implementierungsprozess

03/2014	Finale Entscheidung für GUI Redesign
05/2014	Beauftragung eines Spezialisten für GUIs (Design und Umsetzung)
09/2014	Messe-Präsentation des Prototypen (Dashboard und 4 umgesetzte Menüpunkte)
12/2014	Umsetzung des Facelifts für MVP Start der closed Betaphase für Family & Friends • Start des Nachziehens der selten genutzten Features, dem Finetuning auf Basis des Feedbacks und der Umsetzung von Quickwins



Implementierungsprozess

01/2015	Start der Umsetzung des Facelifts für den restlichen Funktionsumfang Start der public Betaphase • Start des Monitoring der Nutzung des alten und neuen Designs • am Ende der Beta-Phase nutzten 1/3 die neue GUI
03/2015	Neue GUI als Voreinstellung • bei Start dieser Phase nutzten 2/3 die neue GUI E-Mail Befragung der Nutzer
04/2015	Entwicklungsstopp für alte GUI (außer Bugfixing), d.h. neue Features nur in neuer GUI
05/2015	E-Mail Befragung der Nutzer
06/2015	Kommunikation des Abschalttermins (31.08.15) für alte GUI in alter GUI und per Newsletter
08/2015	Abschaltung der alten GUI

Besonderheiten für Lizenzen

- ▶ ein Release pro Quartal, d.h. erhöhter Kommunikationsbedarf, um „Umschalt-Schock“ zu vermeiden
- ▶ Lizenz-Release hinkt SaaS-Version mindestens 2-4 Wochen hinterher (Fokus auf Stabilität statt schnellen Rollouts)
- ▶ zwei Quartale lang kann Lizenznehmer entscheiden, ob er altes oder neues Design nutzen möchte



Tipps zur Umsetzung (1)

- ▶ Nutzung der verschiedenen Features messen, um MVP zu definieren
- ▶ Liste aller umzugestaltenden Seiten erstellen
(wir hatten erst einige übersehen)
- ▶ Browser-Anforderungen so früh wie möglich kommunizieren, damit Nutzer ausreichend Zeit haben, sich darauf einzustellen
- ▶ Intensive Zusammenarbeit zwischen Designern und Entwicklern bei der Planung ist sinnvoll, weil Entwickler i.d.R. nicht gestalten und Designer oft nicht abschätzen können, wie viel Aufwand Ihre Ideen bedeuten.



Tipps zur Umsetzung (2)

- ▶ im Vorfeld Beta-Tester unter interessierten Kunden akquirieren, sonst gibt es in der Beta-Phase praktisch kein Feedback
- ▶ weil Feedback primär in „Zwangssituationen“ erfolgt, diese möglichst früh provozieren (neue GUI als Voreinstellung, Kommunikation des Abschalttermins, etc.)
- ▶ bei hohem Anteil von negativem Feedback nicht beirren lassen, sondern detailliert nachfragen
(jeder 10. beschwert sich, aber nur jeder 100. lobt)

Erfahrungen



Erfahrungen

- ▶ super-konsequent bei der Definition des MVP und beim Zerlegen der Aufgaben in mehrere Schritte sein



Erfahrungen

- ▶ frühes Release des Facelift führt auch zu frühem Feedback, so dass sich Verirrungen reduzieren lassen



Erfahrungen

- ▶ zeitliche Trennung von Facelift, Usability-Verbesserungen und Workflow-Optimierungen erleichtert Nutzern den Umstieg wegen Schritt für Schritt Umgewöhnung



Erfahrungen

- ▶ freiwillig kommt das Feedback der Nutzer nur von den Meinungs-Rändern (Begeisterung oder Verdammnis)



Erfahrungen

- ▶ Ein Soft Launch mit monatelangem Parallelbetrieb vermeidet gestresste Kunden und Zeitdruck



Erfahrungen

- Moderne View-Layer erfordern ein (für Java-Entwickler) völlig neues Knowhow und Tools: Bootstrap, SAAS, jQuery, npm, grunt, ...



Erfahrungen

- ▶ Continuous Delivery ist nach Umsetzung des Facelift-MVP perfekt:
 - Der Kunde gibt eine gute Anregung
 - Anregungen sind innerhalb weniger Tage produktiv
 - Der Kunde ist motiviert („Mit-Entwickler“)



Erfahrungen

- ▶ Ein signifikanter Anteil des Feedbacks kommt erst nach der Ankündigung des Abschalttermins



31.8.–3.9.2015
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence

Vielen Dank für Ihre
Aufmerksamkeit!

31.8.–3.9.2015
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence



Martin Aschoff

AGNITAS AG

maschoff@agnitas.de