

Machbarkeits- analyse



Autoren : Martin Amelsberg (MartinAmelsberg@T-Online.de)
Daniel Finger (Daniel.Finger@gmx.net)
Ralf Schmid (Ralf-Schmid@web.de)

Version : 21

Inhaltsverzeichnis

1. Anforderungen	
1.1. Randbedingungen.....	3
1.2. Funktionale Anforderungen.....	3
2. Qualitätsmerkmale	
2.1. Planning Language.....	4
2.2. Lösungen.....	5
2.3. Wirkungsbewertung.....	7
2.4. Risikobewertung.....	9

1. Anforderungen

1.1. Randbedingungen

- unterstützte Webserver: mindestens Apache, IIS
- unterstützte Betriebssysteme: mindestens Linux, Windows
- Systemvoraussetzungen: WebServer mit PHP (ab Version 3) und MySQL (ab Version 3.23)
- unterstützte Browser: ab Internet Explorer 4, ab Netscape Navigator 4.x
- Schreibrecht auf Dateisystem
- Recht zur Datenbankanlage, Benutzung
- Grundsatz: so viel wie möglich auf Server-Seite, so viel wie nötig auf Client-Seite
- Trennung von Daten und Design

1.2. Funktionale Anforderungen

- Grundsätzliches
 - + Administration ohne HTML-Kenntnisse
 - + automatisches Installationskript (Anlage Datenbank, Grundkonfiguration)
 - + Erweiterbarkeit: Einbindung neuer CSS-Dateien
- User-Bereich
 - + Benutzerregistrierung mit Zusendung des Passworts und der AGB per eMail
 - + Seminaranmeldung nur für registrierte Benutzer
 - + Prüfung der Anmeldeinformationen auf Plausibilität (Termin in der Zukunft, Platz frei)
 - + Eingabe von Kommentaren für besuchte Seminare
 - + Prüfung von Mehrfachanmeldungen (eMail-Adresse wird mehrfach verwendet)
- Admin-Bereich
 - + Anlage und Änderungen an Seminaren durchführen
 - + automatische eMail-Generierung für angemeldete Teilnehmer bei Seminaränderungen
 - + Bearbeitung von Kommentaren durch den Admin
 - + Export der Daten (Anmeldeinformationen, Benutzerdaten usw.) als CSV-Datei
 - + Anpassung der Texte, Farben, Layout, Bilder
 - + schnelle Seminaranmeldung durch den Admin (z. B. bei telefonischer Anmeldung)
- Verwaltung
 - + Information des Admins per eMail bei allen Einträgen
 - + Verwaltung mehrerer Admins und Seminarleiter
 - + Statistik-Funktionen
 - + Informationen über alle registrierten Benutzer
 - + getrennter Bereich für Admin (Layout, Texte) und Seminarleiter (Seminare anlegen, bearbeiten)

2. Qualitätsmerkmale

2.1. Planning Language

Name	QLade
Erklärung	Ladezeit einer Seite mit ISDN-Geschwindigkeit (64 kBit/s)
Maßeinheit	Sekunden bis zur vollständigen Anzeige der Seite
Test	Empirisch
Maximal	30 Sekunden
Geplant	15 Sekunden
Traumwert	< 5 Sekunden

Name	QZugriff
Erklärung	Zugriffsschutz auf Admin- bzw. User-Bereich
Maßeinheit	???
Test	
Maximal	0
Geplant	0
Traumwert	0

Name	QDesign
Erklärung	Einheitliches Aussehen / Layout
Maßeinheit	Anzahl der nicht anpassbaren Seiten in Prozent
Test	Untersuchung Quellcode
Maximal	0
Geplant	0
Traumwert	0

Name	QIndividuell
Erklärung	Individuelle Anpassung möglich
Maßeinheit	Anzahl der veränderbaren Parameter
Test	Untersuchung des WebInterfaces
Minimal	10
Geplant	15
Traumwert	20

Name	QeMail
Erklärung	Dauer bis zur Absendung der Anmelde-eMail
Maßeinheit	Sekunden bis zum Absenden
Test	Repräsentative Anmeldungen, mehrere gleichzeitig → Streßtest
Maximal	30 Sekunden
Geplant	20 Sekunden
Traumwert	5 Sekunden

Name	QEinAdmin
Erklärung	Intuitive Bedienung für den Admin
Maßeinheit	Einarbeitungszeit
Test	Einarbeitungszeit von Testbenutzern
Maximal	30 Minuten
Geplant	20 Minuten
Traumwert	Intuitiv

Name	QEinUser
Erklärung	Intuitive Bedienung für den Benutzer
Maßeinheit	Minuten für die Einarbeitung
Test	Bedienung durch Testbenutzer
Maximal	2 Minuten
Geplant	1 Minuten
Traumwert	Intuitiv

Name	QKorrekt
Erklärung	Korrektheit des Programms
Maßeinheit	Anzahl und Art der Fehlermeldungen in Statistik-Programm
Test	Auswertung Statistik-Programm (verwendet: HitListCommerce Pro 4.0)
Maximal	1 / Sitzung
Geplant	0 / Sitzung
Traumwert	0 / Sitzung

Name	QVerständlich
Erklärung	Verständlichkeit des Codes
Maßeinheit	Minuten
Test	Zeit für Fehlerlokalisierung bzw. Erweiterung (für HTML, PHP, MySQL und CSS Erfahrene Benutzer)
Maximal	20 min
Geplant	10 min
Traumwert	10 min

2.2. Lösungen

Zu jedem Qualitätsmerkmal wird mindestens eine Lösung genannt.

Werden mehrere Lösungen gefunden und/oder waren wir nicht mit der ersten Lösung zufrieden, so notieren wir zu jedem Qualitätsmerkmal mehrere Lösungen, die dann in der Gesamtheit als eine Lösung betrachtet werden.

- QLade:
 - + Auswertungen in MySQL und nicht im Programmcode
 - + Aufbereitung auf Server-Seite

- QZugriff:
 - + Abfrage Benutzername und Passwort für Admin-Bereich
 - + Verwaltung aller Benutzerinfos in einer passwort-geschützten Datenbank
 - + Überprüfung beim Laden jeder Admin-Seite auf gültige Anmeldung und Berechtigung
- QDesign:
 - + Einbindung der im Admin-Bereich änderbaren CSS-Datei in alle Dateien
- QIndividuell:
 - + Anpassung der CSS-Werte über WebInterface
 - + Verwendung vordefinierter CSS-Dateien
 - + Import eigener CSS-Dateien
- QeMail:
 - + synchrone Kopplung von Anforderung und Ausführung
- QEinAdmin:
 - + Erklärende Texte
 - + ToolTipTexte
 - + Vorschauenfenster
 - + einheitliches Layout
 - + Vermeidung von Fachbegriffen
- QEinUser:
 - + Online-Hilfe
 - + ToolTipTexte
 - + Verwendung von Templates
- QKorrekt:
 - + ausführliche Tests
 - + Formale Inspektionen (simplifiziert)
 - + Modularisierung
- QVerständlich:
 - + Modularisierung in „logische Einheiten“
 - + Komplette Auslagerung der PHP-Funktionen in externe Dateien
 - + externe Dateien für mySQL-Befehle
 - + Quellcode-Kommentare

2.3. Wirkungsbewertung

	LLade	LZugriff	LDesign	LIndividuell	LeMail	LEinAdmin	LEinUser	LKorrekt	LVerständlich	Summe1
QLade	80	-10	---	---	---	---	-10	---	20	80
QZugriff	---	100	---	---	---	---	---	10	10	120
QDesign	---	---	100	100	---	---	---	---	---	200
QIndividuell	---	---	100	100	---	---	---	---	---	200
QeMail	---	---	---	---	100	---	---	---	---	100
QEinAdmin	---	-10	20	40	---	100	---	---	---	150
QEinUser	---	-30	20	---	20	---	80	---	---	90
QKorrekt	---	---	10	10	---	-10	-10	100	50	150
QVerständlichkeit	---	-10	10	---	---	---	---	30	90	120
Summe2	80	40	260	250	120	90	60	140	170	

Fazit:

- a) Was bedeuten die Zahlen?

Jede Lösung kann Auswirkungen auf das Erreichen der definierten Qualitätsmerkmale haben. Diese Auswirkungen müssen bewertet werden.

Als Skala wird ein Bereich zwischen -100 (wirkt sich sehr negativ) bis 100 (erfüllt das Qualitätsmerkmal vollständig) verwendet. Beeinflussen sich Anforderung und Lösung nicht, wird das durch „---“ gekennzeichnet.

- b) Wie wirken sich die Lösungen auf das jeweilige Qualitätsmerkmal aus? (Summe1)

Mit 200 Punkten sind *QDesign* und *QIndividuell* am besten gelöst. Dieses Ergebnis kommt zustande, da diese Merkmale sich gegenseitig sehr positiv beeinflussen.

Mit lediglich 80 Punkten ist *QLade* am schlechtesten gelöst. Jedoch wir uns darauf geeinigt, die Zahl zu akzeptieren, da dadurch unser Projekt nicht gefährdet ist.

- c) Wie gut/schlecht sind die einzelnen Lösungen bezogen auf alle Qualitätsmerkmale? (Summe2)

Mit 260 bzw. 250 Punkten wirkten sich *LDesign* und *LIndividuell* am stärksten positiv aus. Folglich werden wir diese Lösungen besonders im Auge behalten.

Mit nur 40 Punkten beeinflusst *LZugriff* am wenigsten positiv. Dies bedeutet aber nicht, dass unsere Lösung schlecht ist.

2.4. Risikobewertung

Entsprechend den Eigenschaftsanforderungen wird das Risiko abgeschätzt. Semantisch heißt das: „Wie hoch ist das Risiko, die Eigenschaftsanforderung Qxxx nicht erfüllen zu können?“

Anmerkung: Bei der Abschätzung wurde die Zeit als Risikofaktor nicht direkt berücksichtigt, da wir keinen festen Zeitplan haben. Zwar haben wir eine ungefähre Vorstellung, doch kommt es uns auf ein paar Tage mehr oder weniger nicht an, da niemand explizit auf unsere Ausarbeitung wartet.

Legende:

0%	→	kein Risiko
...		
50%	→	relativ hohe Risiko
...		
100%	→	sehr großes Risiko
QLade	:	42%
		+ Erfahrung im Webdesign fehlt
		+ Wegen Individualisierung des Admins nicht direkt abschätzbar
		+ Zugriffszeit der Datenbank noch unbekannt
QZugriff	:	10%
		+ viele Quellen diesbezüglich im Internet
		+ Zentrale Verwaltung der Design-Elemente
QDesign	:	10%
		+ Seiten haben prinzipiell immer die gleichen Themen
QIndividuell	:	≈0%
		+ eines der Projekt-Hauptziele
QeMail	:	≈0%
		+ Modul bereits aus Vorgängerprojekten bekannt
QEinAdmin	:	20%
		+ Beschreibungen und Dokumentationen sind individuell zu bewerten
		+ es gibt unerfahrene Admins

- QEinUser : 20%
- + Beschreibungen und Dokumentationen sind individuell zu bewerten
 - + es gibt „dumme“ User
 - + kein Programm kann 100 % fehler-freundlich sein
- QKorrekt : 20%
- + „Jeder Mensch macht Fehler“
 - + Komplexität des Projektes
- QVerständlich : 20%
- + Beschreibungen und Dokumentationen sind individuell zu bewerten
 - + kein C :-)
 - + modularisierte Funktionen werden in logische Themenpaketen zusammengefasst