

Stark verkürzter Auszug Dokumentation zur betrieblichen Projektarbeit

Automatische Datentransformation

Permanenter automatischer Datenabgleich des innerbetrieblichen Datenbestands mit der Website des Unternehmens

1. Einleitung

1.1. Projektziel

Ziel des Projekts ist die automatische Aktualisierung der Datenbank auf der Webseite des Auftraggebers. Um dies zu erreichen müssen die Daten von dem innerbetrieblichen SQL-Server durch eine eigenständige Desktopanwendung mit grafischer Benutzeroberfläche (GUI) und eigener Datenbank täglich eingelesen und in eine Access-Datenbank exportiert werden. Diese werden dann durch einen FTP-Tunnel auf den, vom Webentwickler erstellten Dateiserver, hochgeladen.

Ferner soll die Anwendung dem Benutzer ermöglichen mehrere Exportaufgaben und deren Einstellungen (Dauer, Ausführungsuhrzeit und etc.) selbst zu steuern.

1.2. Ausgangslage

Die Ausgangslage des Projekts bildet einige grundsätzliche Probleme beim Auftraggeber, die im Betrieb der Datenübermittlung ein erhebliches Optimierungspotential aufweisen.

Zum einen liegt die Website im Ausgangszustand auf einem dedizierten Server, die innerbetriebliche Datenhaltung hingegen auf einem SQL-Server. Zusätzliche Dateien (Cover, PDFs mit Inhaltsverzeichnissen, Leseproben etc.) liegen auf einem Dateiserver. Diese sind mit den Datensätzen von einzelnen Titelstammsätzen verknüpft. Jede Datenänderung an einem Titel-Stamm wird manuell auf dem Webserver gepflegt bzw. aktualisiert. Diese Vorgehensweise ist zeitaufwändig, fehleranfällig und kostspielig. Außerdem passiert es oft, dass durch die manuelle Eingabe der Daten einzelne Prozessabschnitte vergessen werden.

Ein weiteres Problem ist, dass die Daten aus CMS nicht jederzeit aktualisierbar sind. Sie werden lediglich einmal pro Woche durch extern beauftragtes Personal übertragen. Das CMS ist zudem aus Sicherheitsgründen nur im Haus erreichbar, was die Datenübermittlung an den Standort des Unternehmens bindet.

Aufgrund dieser vorherrschenden Situation und im Zuge der Automatisierungspolitik des Auftraggebers hat sich der SV dazu entscheiden, die Entwicklung einer Anwendung im Auftrag zu geben, durch die der gesamte Prozess automatisiert werden soll.

2. Analysephase

1.3. Ist-Analyse

In mehreren Gesprächen mit der zuständigen Person des SV zum Datenaustausch wurden die bisherigen Publikationsaufwände ermittelt. Die zu übertragenden Daten (bestehend aus Titeln, Autoren und CMS-Einträgen) werden momentan einmal pro Woche der o.g. Person per Mail übermittelt. Diese werden dann, in Form einer Access-Datenbank, in einer vom Webentwickler vorgeschriebenen Infrastruktur exportiert und auf den Dateiserver beim Webhosting-Provider hochgeladen. Zuletzt wird das Importverfahren angestoßen, indem eine bestimmte Webseite aufgerufen wird. Die Seite und deren Funktionen wurden ebenfalls von mir implementiert, jedoch wird dies an dieser Stelle nicht weiter betrachtet.

1.4. Wirtschaftlichkeitsanalyse

Ausgehend von dem Automatisierungsziel des SV und aufgrund der zu optimierenden Praxis der Datenübermittlung (vgl. Abschnitt 1.3 und 3.1), ist die Umsetzung des Projekts unbedingt erforderlich, da ein automatisierter Datentransfer dem SV deutliche Vorteile bringt. Inwieweit die Realisierung aber auch unter wirtschaftlichen Gesichtspunkten gerechtfertigt ist, soll in den folgenden Abschnitten aufgezeigt werden.

1.4.1. „Make or Buy“-Entscheidung

Da es sich bei dem Projekt um eine unternehmensspezifische Anforderung des SV handelt, ist auf dem Software-Markt keine Lösung verfügbar, die diese Aufgabe erfüllen könnte. Daher soll das Projekt in Eigenentwicklung durchgeführt werden.

1.4.2. Projektkosten

Für die Projektkosten, die während der Entwicklung aufkommen, muss neben den Personalkosten auch noch der Kostensatz für die Arbeitsplatzausstattung pro Stunde berücksichtigt werden.

Da die genauen Personalkosten an dieser Stelle nicht herausgegeben werden dürfen, wird die Kalkulation anhand der Stundensätzen durchgeführt, die vom Entwicklungsteamleiter festgelegt werden. Seinen Angaben zufolge, beträgt der Stundensatz eines Auszubildenden 10€ und eines Mitarbeiters 30€.

Für die Berechnung wird ein pauschaler Stundensatz von 10€ für die Ressourcennutzung angenommen. In Abbildung 2 lässt sich die Kalkulation der Kosten der einzelnen Vorgänge nachvollziehen.

Vorgang	Mitarbeiter	Zeit	Personal	Ressourcen	Gesamt
Entwicklung	1 x Azubi	59 h	648 €	648 €	1296 €
Fachgespräch	1 x Azubi 1 x Mitarbeiter	6 h	60 € 180 €	120 €	360 €
Test	1 x Azubi	2 h	20 €	20 €	40 €
Abnahme	1 x Azubi 1 x Mitarbeiter	1 h	10 € 30 €	20 €	60 €
				Summe	1756 €

Tabelle 1: Kostenaufstellung

1.4.3. Amortisationsdauer

Im folgenden Abschnitt soll ermittelt werden, ab welchem Zeitpunkt das Investitionsprojekt seine Anschaffungsausgaben refinanziert hat. Anhand dieses Wertes kann dann beurteilt werden, ob die Umsetzung des Projektes aus wirtschaftlicher Sicht sinnvoll ist und sich auf Dauer Kostenvorteile ergeben.

Beispielsweise können sich die Personalkosten reduzieren.

Eine Übersicht über die konkrete Kosten- und Zeiteinsparung der einzelnen Vorgänge wird in Abbildung 3 dargestellt. Die für die Vorgänge angegebenen Zeiten wurden mir von der Personalabteilung des SV (bei der alten Vorgehensweise) mitgeteilt.

Vorgang	Anzahl pro Monat	Zeit (alt) pro Vorgang	Kostenaufwand (alt) Pro Stunde	Zeit (neu) pro Vorgang	Einsparung pro Monat
Sendung der gewünschten Änderungen	8	20 min	30 €	0	80 €
Datenexport und hochladen durch das externe Personal	4	90 min	50 €	0	300 €
Zeiteinsparung pro Monat					380 €

Tabelle 2: Zeiteinsparung

Nachfolgend soll nun der Return On Investment ROI berechnet werden. Der ROI gibt an, ab welchem Zeitpunkt die Anschaffungskosten durch die Zeitersparnis beglichen werden können.

Außerdem sollte die Differenz der monatlichen Kosten des Webservers in Betracht gezogen werden, da durch die Umsetzung der Anwendung kein dedizierter Server erforderlich sind. Die Kosten werden dadurch von 160 €/Monat³ auf 6,99 €/Monat⁴ reduziert und ergibt eine Einsparung von 153,01 €/Monat. Demzufolge wird samt der in Abbildung 3 dargestellten Einsparung der Kostenersparnis (380 €/Monat) insgesamt 533,01 €/Monat gespart.

Anhand der ROI-Rechnung ergibt sich für das Projekt eine Amortisationsdauer von ca. 3 Monate und 10 Tage. Die neue Anwendung muss mindestens über diesen Zeitraum im Einsatz sein, damit sich Anschaffungskosten und Kosteneinsparung ausgleichen.

Da das Unternehmen die neue Anwendung jedoch langfristig einsetzen wird, kann das Projekt auch unter wirtschaftlichen Gesichtspunkten als rentabel eingestuft werden.

1.5. Anwendungsfälle

Um den Anwendungsfällen eine grobe Übersicht zu geben, wurde in der Zeit der Analysephase ein Use-Case-Diagramm erstellt, welches im Anhang A.3 gezeigt wird. Dieses Diagramm enthält sämtliche Funktionen, die der Endverbraucher benötigt.

3. Entwurfsphase

1.6. Zielplattform

Vorliegendes Abschlussprojekt soll als eigenständige Desktopanwendung umgesetzt werden. Hierbei soll das Datenbanksystem MSSQL zum Einsatz kommen, da sich dieses seit langem im Unternehmen bewährt hat. Für die Umsetzung des Projekts wird die Programmiersprache C# verwendet.

1.7. Architekturdesign

Das Projekt soll auf Basis des Model View Controller (MVC)-Architekturmusters umgesetzt werden. Damit lässt sich jede Komponente einer Software einem der drei Bestandteile (Model, View oder Controller) dieses Musters zuordnen.

Jeder dieser drei Teile hat einen speziellen Aufgabenbereich, der von denen der anderen weitestgehend unabhängig ist. Das Model setzt sich hierbei folgendermaßen zusammen:

- die Daten der entsprechenden Verarbeitungslogik
- der View für die Anzeige der Daten
- der zuständige Controller für die Steuerung der Anwendung. Er stellt das Bindeglied zwischen Model und View dar.

Für die Implementierung und Gestaltung der Benutzeroberfläche soll das Windows Form .NET Framework verwendet werden, welches mit zwei Komponenten, dem Form Designer (View in MVC) und dem Code File (.cs) (Controller in MVC) arbeitet.

Das Entity-Framework wird verwendet, um eine automatische Interaktion mit dem SQL-Server zu ermöglichen. Damit werden dem Entwickler diverse Funktionen zur Verfügung gestellt, mit denen sich die Objekte der Anwendung mit den Tabellen der Datenbank verbinden.

Für die selbstständige Erstellung von Access-Datenbanken wird die ADOX Erweiterung von Microsoft verwendet. ADOX (ActiveX Data Objects) ist eine Erweiterung von ADO Objects und Programmierungsmodell. ADOX beinhaltet Objekte für Erstellung bzw. Modifizierung sowie Sicherheit eines Datenbankschemas

1.8. Entwurf der Benutzeroberfläche

Es wurde zunächst ein Prototyp von der Oberfläche mit Hilfe von Mockups erstellt. In der Hauptansicht soll in der oberen Hälfte eine Liste von Exportaufgaben und in der unteren Hälfte einen kurzen Bericht von ausgeführten Aufgaben mit der Angabe der Zeit und entsprechender Meldung, ob die Aufgabe erfolgreich ausgeführt wurde, angezeigt werden.

1.9. Geschäftslogik

werden Durch die Vorgehensweise der Testerfahrungen während der Entwicklung ergeben sich die zu implementierenden Klassen erst während der Implementierungsphase.

Die Erstellung eines Klassendiagramms ist daher im Voraus noch nicht möglich.

1.10. Pflichtenheft

Am Ende der Entwurfsphase wurde ein Pflichtenheft erstellt. Dieses beschreibt wie und womit der Autor die Anforderungen des Auftraggebers umsetzen möchte. Es dient somit als Leitfaden für die Realisierung des Projektes.

4. Implementierungsphase

1.11. Implementierung der Benutzeroberfläche und Geschäftslogik

Für die Implementierung von der Benutzeroberfläche und Geschäftslogik wurde Visual Studio Community 2015 verwendet.

Nach der Erstellung des Windows-Form-Projektes wurde das Main-Form als Hauptansicht betrachtet, die in Abschnitt 4.3 definierte Benutzeroberfläche für das Hauptfenster hinzugefügt und schrittweise nach dem Iterationsplan die Funktionen, Klassen und Formulare implementiert.

Die Benutzeroberfläche wurde auf Grundlage der in der Entwurfsphase erstellten Mockups entworfen. Die Funktionalität, die sich hinter den einzelnen GUI Elementen verbirgt, wurde über das Code File (.cs) festgelegt. In diesem wurden die Funktionen, die über die anderen Klassen der Anwendung zur Verfügung gestellt werden, den entsprechenden Elementen zugeordnet. Da die Funktionsweise dieser Methoden bereits in der Entwicklungsphase getestet wurde, erfolgte an dieser Stelle nur noch ein Integrationstest. Dieser stellt den korrekten

Ablauf und Aufruf der einzelnen Funktionen sicher. Die gesamte Implementierungsphase konnte mit dem Fertigstellen der Benutzeroberfläche abgeschlossen werden

5. Abnahmephase

Nach der Implementierung sämtlicher neuer Programmmodule und Abschluss aller notwendig gewordenen Veränderungen, wurde das gesamte System unter folgenden Gesichtspunkten getestet:

- a) Funktionalität
- b) Fehleranfälligkeit.

Da bereits während der Programmierung permanent Testsituationen simuliert wurden, und somit aufkommende Fehler bereits behoben werden konnten, entfielen in dieser Phase nachträgliche Korrekturen.

Das gesamte Projekt wurde mit Hilfe eines Setup-Projekts bereitgestellt. SetupProjekte werden zur Erstellung von Windows Installer-Dateien (MSI-Dateien) verwendet, über die Anwendungen für die Installation auf einem anderen Computer oder Webserver verteilt werden. Zwei Arten von Setup-Projekten können erstellt werden. Über herkömmliche Setup-Projekte werden Installationsprogramme erstellt, die Windows-Anwendungen auf einem Zielcomputer installieren

6. Einführungsphase

In Zusammenarbeit mit dem Entwickler der betriebsinternen Anwendung des SV wurden folgende Punkte im Vorfeld abgeklärt.

- ✓ Begriffsdefinitionen aus dem spezifischen Umfeld des Kunden und deren Darstellung in Tabellen und Spalten auf dem SQL-Server
- ✓ Welche Daten/Tabellen werden für den Export zur Verfügung gestellt?
- ✓ Welche Dateninhalte und Relationen von der Datenquelle müssen in der Datenbank auf dem Webserver wiedergegeben werden?

7. Dokumentation

Die Dokumentation der Übertragungswertverwaltung hat drei Bestandteile:

1. die Projektdokumentation,
2. das Benutzerhandbuch,

In der Projektdokumentation werden die einzelnen Phasen beschrieben die während der Umsetzung des Projektes durchlaufen wurden.

Das Benutzerhandbuch enthält Informationen über den Aufbau und die Funktionsweise der Anwendung. Ein Auszug aus dieser Dokumentation befindet sich im Anhang A.13.

Eine detaillierte Beschreibung der entwickelten Klassen befindet sich in der Entwicklerdokumentation. Diese Dokumentation soll dem Entwickler als Übersicht und Nachschlagewerk dienen. Mit Hilfe des Tools Sandcastle wurde diese Dokumentation automatisch generiert. Dies Erstellt aus Quellkomponenten übersichtliche Klassenbibliotheks-Dokumentationen im MSDN-Stil und integriert optional auch Kommentare aus XML-Dokumentationen

8. Fazit

8.1 Soll-/Ist-Vergleich

Beim Rückblick des IHK-Abschlussprojektes wird festgestellt:

- Alle zuvor festgelegten Anforderungen gemäß dem Pflichtenheft wurden erfüllt
- Der zu Beginn des Projektes im Abschnitt 2.1 Projektphasen erstellte Projektplan konnte eingehalten werden

Projektphase	Soll	ist	Differenz
Analyse	16 h	16 h	0 h
Entwurf	6 h	6 h	0 h
Realisierung	30 h	25 h	- 4 h
Test und Abnahme	3 h	2 h	- 1 h
Dokumentation	13 h	19 h	6 h
Gesamt	68 h	69 h	1 h

Von der Zeitplanung wurde insgesamt 1 Stunde abgewichen. Die sich daraus ergebenden Differenzen konnten untereinander kompensiert werden. Das Projekt konnte somit in dem von der IHK festgelegten Zeitrahmen von 70 Stunden umgesetzt werden.

8.2 Ausblick

Durch die erfolgreiche Durchführung des Projektes und des zufriedenen Kunden konnten weitere Bearbeitungsbereiche generiert werden:

- Die Realisierung der im Lastenheft definierten Anforderungen bildet eine Grundlage für weitere zukünftige Entwicklungen.
- Der Relaunch der Website mit weiteren Dateninhalten ist bereits in Auftrag gegeben.
- Die Anwendung soll auch auf der Website eines Schwesterunternehmens zur Anwendung kommen.
- Aufgrund des im Abschnitt 4.2 (Architekturdesign) beschriebenen modularen Aufbaus des Projektes können Anpassungen bzw. Erweiterungen sehr einfach vorgenommen werden. Die Modularität der Anwendung ermöglicht somit eine gute Wartung und Erweiterbarkeit.

Somit ist angedacht dieses Projekt bei weiteren Entwicklungen im Hause Q-Sys für weitere Entwicklungen bei anderen Kunden hinsichtlich der zukünftigen Wege Richtung Wirtschaft 4.0 global einzusetzen..

Datenbankschema dedizierter Server	ist eine formale Beschreibung der Struktur von Daten. Ein dedizierter (englisch dedicated [...], dt. wörtlich gewidmet, bestimmt oder zugeordnet) Server ist ein Server (eigenständiges Gerät oder in einer Baugruppe mit Merkmalen eines eigenständigen Gerätes), der für einen Netzwerkdienst oder mehrere dauerhaft betriebene Dienste vorgesehen ist, und nicht nebenbei für andere Aufgaben, etwa als Workstation, genutzt wird. books.google.de Titel: Einführung in die PC-Grundlagen Von: Jürgen Ortmann
Desktopanwendung	wird ein Anwendungsprogramm bezeichnet (kurz „Anwendung“, engl. „application software“) und ist ein Computerprogramm, das Benutzer anwenden, um eine nützliche Funktion zu erreichen.[1]
Entity-Framework Iterationsplan	kurz auch EF, ist ein Framework für objektrelationale beschreibt allgemein einen Prozess mehrfachen Wiederholens gleicher oder ähnlicher Handlungen zur Annäherung an eine Lösung oder ein bestimmtes Ziel
Mockup	(auch Maquette) bezeichnet im Deutschen beispielsweise eine Attrappe. Er wird heute aber meist für ein maßstäblich gefertigtes Modell bzw. eine Nachbildung zu Präsentation Zwecken benutzt, demgegenüber ist der Prototyp ein funktionsfähiges Modell.
Prototyp	stellt in der Technik ein für die jeweiligen Zwecke funktionsfähiges, oft aber auch vereinfachtes Versuchsmodell eines geplanten Produktes oder Bauteils dar.
Return On Investment	bezeichnet ein Modell zur Messung der Rendite einer unternehmerischen Tätigkeit
SQL-Server	(auch kurz MSSQL-Server) ist ein relationales Datenbankmanagementsystem von Microsoft.
Webhosting	Unter Webhosting versteht man die Bereitstellung von Webspace sowie die Unterbringung (Hosting) von Webseiten auf dem Webserver eines Internet Service Providers (ISP)
Webserver	Als Webserver bezeichnet man den Computer mit Webserver-Software oder nur die Webserver-Software selbst. Webserver werden lokal, in Firmennetzwerken und überwiegend als WWW-Dienst im Internet eingesetzt.
Wirtschaft 4.0	In der Industrie 4.0 verzahnt sich die Produktion mit modernster Informations- und Kommunikationstechnik. Treibende Kraft dieser Entwicklung ist die rasant zunehmende Digitalisierung von Wirtschaft und Gesellschaft. Sie verändert nachhaltig die Art und Weise, wie zukünftig produziert und gearbeitet wird: Nach Dampfmaschine, Fließband, Elektronik und IT bestimmen nun intelligente Fabriken (sogenannte „Smart Factories“) die vierte industrielle Revolution.

Abkürzungsverzeichnis

ADOX ActiveX Data Objects

FTP File Transfer Protocol

GUI graphical user interface

MSSQL Microsoft SQL Server, Microsoft SQL Server

MVC Model View Controller Q-Sys Q-Sys Business e.K.

ROI Return On Investment

SSMS SQL Server Management Studio, SQL Server Management Studio

SV Südverlag

UML Unified Modeling Language

XML Extensible Markup Language (engl. „erweiterbare Auszeichnungssprache“)