

Studiengang Informatik der ZHAW am Standort Zürich

Verfassen und Vorlage einer Kurzfassung

1. Allgemeines

Die Kurzfassung ist eine (separate) Beilage zur Bachelorarbeit im Studiengang Informatik. Sie fasst die Bachelorarbeit für Interessierte so zusammen, dass sie veröffentlicht werden kann, z. B. in Jahresberichten oder auf der Homepage der ZHAW. Damit ist die Kurzfassung ein eigenständiger Text. Die Kurzfassung ist als pdf- oder Word-Datei bei der Abgabe der schriftlichen Arbeit mit abzugeben.

2. Äussere Form

Der Umfang beträgt eine A4-Seite mit total ca. 4500 Zeichen (mit Leerzeichen; ohne Abbildung). Eine Kurzfassung mit Abbildung besteht aus ca. 3200 Zeichen.

Die Kurzfassung enthält die Abschnitte Ziel, Grundlagen, Vorgehensweise, Ergebnisse und allenfalls die wichtigsten Quellenangaben der Bachelorarbeit (max. 3). Die Abschnitte gliedern die Kurzfassung ohne eigene Überschriften.

Über der Kurzfassung stehen der Titel der Bachelorarbeit, Autorin(nen)/Autor(en), Hochschule und Studiengang sowie das Datum der Abgabe. Die Abbildung ist beschriftet (Abb.-Text unterhalb der Abbildung; Legende) und möglichst selbsterklärend.

3. Stil

Stil und Ausdrucksweise folgen den üblichen Standards zum Verfassen technischer Berichte [1]. Die Kurzfassung ist auf Hochdeutsch und im Präsens verfasst. Der Text ist ausformuliert (keine Aufzählungen, Listen oder Tabellen). Alle Abkürzungen und Akronyme sind beim ersten Auftauchen im Text zu erläutern. Beispiele sollen sehr kurz gehalten werden.

Der Inhalt ist geschlechtergerecht [2] und so zu verfassen, dass beteiligte Firmen und Personen nur anonymisiert auftauchen.

4. Beispiel und Vorlage

Das Beispiel dient als Muster und Vorlage für eine Kurzfassung ohne Abbildung. Eine allfällige Veröffentlichung der Kurzfassung wird übertragen und neu formatiert.

5. Support und Literatur

Jede Kurzfassung wird vor Abgabe in Zusammenarbeit mit Dr.-Ing. Ralf Mock, Leiter aF&E: IT Sicherheit und Zuverlässigkeit, oder Christian Walser verfasst. Dieses Dokument und die angegebenen Quellen sind über sie erhältlich.

Lagerstrasse 45
Büro ZL O3.15
8021 Zürich

Tel.: 058 934 82 57
mock@zhaw.ch

[1] Mock, R., Verfassen eines technischen Berichtes (Bachelorstudiengang Informatik: Deutsch). Zürich, HSZ-T, 2010

[2] *Geschlechtergerechte Sprache. Leitfaden zum geschlechtergerechten Formulieren im Deutschen*, Schweizerische Bundeskanzlei, in Zusammenarbeit mit der Zürcher Hochschule für Angewandte Wissenschaften. Bern, 2009 siehe auch: www.bk.admin.ch/dokumentation/sprachen/04915/

Integration einer Netzwerktopologie einer Support-Plattform

xxx Maier

Hochschule für Technik Zürich, Studiengang Informatik (1. April 2010)

Ein Kabelnetz ist eine baumartige Struktur, die ein Signal oder Datenpakete über einen Rundruf (Broadcast) von einem Wurzelknoten (Node) zu den Teilnehmern überträgt. Das auftraggebende Unternehmen dieser Arbeit speicherte die Pläne seiner Netze (Liegenschaften, Signalübergangsstellen usw.) bisher mit einem proprietären CAD-Programm (Computer-aided Design) ab, was als einzige Zugriffsmöglichkeit das manuelle Öffnen der Netz-Schemata bot. Mit einer neuen Version des Programmes können jedoch auch die Verbindungen der Komponenten untereinander über eine Datenbank abgerufen werden. Diese Arbeit erschliesst das Potential der neu zugänglichen Informationen, bereitet diese für betriebliche Anwendungen auf und stellt sie über einen Service zur Verfügung.

Eine methodengestützte Anforderungsanalyse unterstützt die Software-Entwicklung, um die für das auftraggebende Unternehmen passende Programmiersprache zu finden. Für die Software-Entwicklung standen drei Methoden zur Diskussion: Use Case, Requirements Engineering nach IEEE 830 und User Stories. Die Methode der User Stories, die sich durch Einfachheit und intensive verbale Kommunikation mit den Benutzern auszeichnet, wird schliesslich eingesetzt. Eine SWOT-Analyse (Strengths, Weaknesses, Opportunities, Threats) ergibt als Programmiersprache Java und für die Middleware einen SOAP-Webservice (Simple Object Access Protocol). Diese passen somit am besten zur Unternehmensstrategie und zu den Mitarbeitenden. Funktionale Anforderungen sowie Anforderungen an die Service-Performance (Schnelligkeit der Abfragen, Reaktionszeit u. a.) und an die Schnittstellen werden ebenfalls bestimmt.

Der implementierte Service legt die aufbereiteten Daten auf einem Datensystem ab und lädt diese bei Bedarf. Bei hinreichend kleinen Dateien bleibt damit die Ladezeit kurz genug – Suchoperationen im Netzwerk erfolgen rasch im Arbeitsspeicher. Die Service-Architektur teilt sich in einen Import- und Service-Teil auf. Der Import-Teil ist eine Java-Applikation. Die Konsumenten reichern die Topologie-Informationen an, beispielsweise mit Kundendaten, und verarbeiten sie. Der für diese Arbeit entworfene Client stellt die Informationen über Webseiten zur Verfügung. Java-Klassen modellieren Topologie-Informationen und werden über einen ObjectOutputStream in eine Datei serialisiert. Ein Composite-Pattern stellt die Beziehungen zwischen den Netzwerk-Elementen dar. Die Traversierung, d.h. die Bestimmung der Route durch Knoten und Kanten in einem baumartigen Graphen, folgt einer in der Informatik üblichen Art, wobei in dieser Arbeit die Tiefensuche gute Resultate liefert. Die Implementierung des Services besteht aus drei Teilen. Der erste Teil bildet den Kern der Applikation, da er das Domänenmodell und den Datenzugriff beherbergt (Logik und Datenrepräsentation). Der zweite Teil ist ein Kommandozeilen-Tool, das entweder sämtliche Nodes oder nur eine Auswahl davon importiert. Das Ergebnis sind zwei Ordner, wobei einer den Index über sämtliche erfassten Adressen plus Projektnummern enthält. Der andere Ordner beinhaltet die serialisierten Klassen. Im dritten Projektteil werden die vorigen Teile zusammengeführt. Ein Spring-Webservice setzt das definierte Interface um und greift auf diesen mit einem zweiten Servlet (einer JSF-Webapplikation) zu. Ein weiteres implementiertes http-Servlet dient dazu, Netzwerk-Schemata als Bilddateien zu erhalten.

Mit dem Prototyp des implementierten Services kann das auftraggebende Unternehmen Informationen aus Netzwerk-Schemata besser nutzen. Allerdings zeigte der erste Anwendungs-Durchgang, dass kaum Liegenschaften gefunden wurden, die abgebildete Netzwerk-Topologie somit unvollständig war: Das Datenmodell des Ursprungsystems speichert jene Liegenschaften, die an eine Signalübergangsstelle in einem anderen Haus angeschlossen sind, nicht in der Datenbank ab. Weil aber fast alle Signalübergangsstellen noch andere Häuser mit Signalen speisen, blieb fast die Hälfte der tatsächlich am Kabelnetz angeschlossenen Liegenschaften unentdeckt. Nach der Eingabe bereits manuell erfasster Informationen liefert der entwickelte Service brauchbare Resultate. Eine solche Vervollständigung der Informationen erlaubt dann auch komplexere Topologie-Abfragen.