



Staatlich geprüfte(r) Technische(r) Assistent/in Informationstechnik

Praktikumsbericht

Firma: Code One GmbH

Thema:

Bericht Nr.: 1

Name: David Brüggemann

Klasse / Gruppe: IF3A Gr.B

Mitarbeiter: ---

Abgabedatum: Olsberg, 02.05.2011

Fachlehrer: Herr Kramer-Gerke

Bewertung:

Inhaltsverzeichnis

1	Einleitung.....	2
1.1	Betrieb.....	2
1.2	Produkt.....	2
1.3	Prinzip.....	2
2	Mein Projekt.....	3
2.1	Allgemeine Funktion.....	3
2.2	Vorgaben.....	3
2.3	Durchführung	4
2.3.1	Erste Aufgabe.....	4
2.3.2	Zweite Aufgabe.....	5
2.4	Hilfsmittel.....	8
3	Wertung.....	8
4	Quellen.....	9

Abbildungsverzeichnis

Abbildung 1:	Logo der Code One GmbH.....	2
Abbildung 2:	Backpack Kit mit angeschlossener Kamera.....	2
Abbildung 3:	Prinzip des Backpack kits von Code One.....	2
Abbildung 4:	Anfangszustand: So habe ich die Webcontrol zunächst an die Hand bekommen	3
Abbildung 5:	Zwischenstand: So sah die Webcontrol nach Abschluss der ersten Aufgabe aus	4
Abbildung 6:	Finale Version: Nach der zweiten Aufgabe war meine Bearbeitung soweit komplett. (Seite in Deutsch).....	7
Abbildung 7:	Logo wikibooks: Javascript.....	8
Abbildung 8:	offizielles Logo: Notepad++.....	8
Abbildung 9:	offizielles Logo: GIMP.....	8

1 Einleitung

1.1 Betrieb

Mein Berufspraktikum habe ich in der Code One GmbH absolviert. Dieses noch sehr junge Unternehmen beschäftigt sich mit dem Live Streamen von Videoaufnahmen. Die Firma wurde im Februar 2010 gegründet und hat ihren Firmensitz in Düsseldorf. Es ist ein Inhabergeführtes Unternehmen. Das Ziel der GmbH ist es mobiles Live Video Streaming einfach, flexibel und zuverlässig wie möglich zu gestalten und das egal wo man sich gerade befindet.



Abbildung 1: Logo der Code One GmbH

1.2 Produkt

Die Firma entwickelt verschiedene Gerätschaften um dieses Videostreamen zu erleichtern doch das Hauptaugenmerk liegt auf dem sogenannten Backpack Kit. Dieses Produkt hat viele



Abbildung 2: Backpack Kit mit angeschlossener Kamera

Bezeichnungen, sie meinen aber alle das selbe. Dieser auch genannte Router beinhaltet einen Encoder, der das eingehende Videosignal bearbeitet und gegebenenfalls auch teilt. Zudem enthält dieser Encoder auch neben den verschiedenen Anschlüssen für Videosignale, USB und Ähnlichem mehrere sogenannte Module. Diese ermöglichen das Versenden des encodierten Videosignals ins Internet über DSL (Ethernetanschluss) oder auch über das Handynetz (UMTS). Gesteuert wird das Produkt über die selbst entwickelte Software durch ein kleines Touchdisplay, welches ca. 7 Zoll groß ist. Um die Mobilität des Backpacks zu gewährleisten, kann es durch 2 handelsübliche Kameraakkus betrieben werden. Dies ist, nebenbei erwähnt, zur Zeit noch einzigartig.

1.3 Prinzip

Das Prinzip des Live Videostreamens von Code One ist recht simple. Man startet den „Rucksack“ und schließt eine oder mehrere Kameras an das Gerät. Das Backpack encodiert das eingehende Signal live zu h264 und splittet es auf bis zu 6 Signalteile auf. Diese werden danach über die Module per UMTS oder DSL an ein Rechenzentrum geschickt, welches die Signalteile decodiert und wieder zusammenführt. Nun ist das Videosignal der Kamera wieder komplett und kann von jeder beliebigen Webseite aus aufgerufen werden.



Abbildung 3: Prinzip des Backpack kits von Code One

2 Mein Projekt

Neben einigen andern Projekten habe ich ein größeres Projekt an die Hand bekommen welches sich Webcontrol nennt.

2.1 Allgemeine Funktion

Die Webcontrol, die ich in meinem Praktikum bearbeitet habe, ermöglicht eine Steuerung von Module, die zum Verschicken eines Videosignals eines Routers von Code One benutzt werden. Um bestimmte Einstellungen vor Ort oder auch per Fernzugriff zu ermöglichen wird die Webcontrol dazu eingesetzt, das Verhalten der Module über einen Webbrowser zu steuern.

2.2 Vorgaben

Die Webseite besaß zunächst weißen Hintergrund auf dem verschiedene „Tunnel“ zu sehen waren. Tunnel beschreiben die Verbindung der 6 Module eines Routers zum Rechenzentrum. Über diesen Tunnel „läuft“ quasi das Videosignal. Jeder der Tunnel besitzt eine Bezeichnung, die über dem jeweiligen Tunnel in der Webcontrol angezeigt werden. Direkt darunter zeigen 6 Graphen den Zustand der dazugehörigen Module des Routers. Beschrieben werden dort graphisch die Up und Download werte und die Latenz des Videosignals. Die Latenz, Latenzzeit oder auch Verzögerungszeit beschreibt den Zeitraum zwischen einem verborgenen Ereignis und dem Eintreten einer sichtbaren Reaktion darauf. Die Latenz wird somit in Millisekunden angegeben. Der aktuelle Wert der Latenz aber auch der Up- und Download, dessen Einheit in kbit/s angegeben sind, kann jeweils neben den Graphen betrachtet werden. Zusätzlich sind die Graphen nochmals mit einer eigenen Überschrift beschriftet. Die Hyperlinks, die sich unter den Graphen befinden, sind die Steuerelemente der Module. Sie senden Befehle über das Netzwerk an den Encoder der diese interpretiert und ausführt. Dabei kann man beispielsweise einzelne Module an oder ausstellen oder einstellen, dass die Latenz eines Moduls sich nicht über einen Wert befinden darf. Die gesamte Webseite wird von einer „control.exe“ erstellt. Diese nimmt übernimmt den Inhalt zunächst zweier txt-Dateien, in denen Header und Schluss der Webseite in HTML-Code stehen. Der restliche Inhalt der Seite wird aus 2 Javascript Dateien „gebaut“. Dabei beinhaltet die eine Datei den Code für die Graphen und die andere wird für den Aufbau der Seite, wie Ausrichtung und Beschriftung gebraucht. Die eigentliche Beschriftung der Befehle stehen allerdings in einer „settings.ini“, in der auch Informationen und Daten zum den Tunneln stehen. Quasi muss nur diese Datei verändert werden, damit die Webcontrol die gewünschten Befehle abschickt oder die gewollten Tunnel anzeigt.



Abbildung 4: Anfangszustand: So habe ich die Webcontrol zunächst an die Hand bekommen

2.3 Durchführung

2.3.1 Erste Aufgabe

Da ich somit die Webcontrol in einem recht schlichten, nicht benutzerfreundlichen Aufbau bekommen habe, war meine erste Aufgabe die Seite allgemein zu verschönern. Außerdem sollten die Elemente der Seite nebeneinander ausgerichtet werden, so dass die Hyperlinks, die die Befehle versenden, neben den Graphen zu sehen sind. Quasi sollen also die Tunnel nebeneinander stehen und sich nach rechts ausbreiten statt nach links. Dies sollte mehr Übersicht schaffen. Zunächst überlegte ich mir eine neue Hintergrundfarbe. Dazu wählte ich einen Verlauf von dunkel-orange zu transparent (also weiß), um die typischen Farben der Firma anzudeuten. Die Hyperlinks ordnete ich wie gewünscht von den Graphen aus rechts zu. Dies dabei halfen mir einige selbst erstellte CSS-Regeln. CSS ist ein sogenannter Stylesheet für HTML-Seiten der hauptsächlich dazu dient diese effizient zu verschönern. Neben der Erstellung dieser Regeln musste ich in dem vorhandene Javascript-Dokument, welchen für das Aussehen der Seite zuständig ist, in der Reihenfolge der Aufrufe leicht verändern. Dazu war es nötig zu verstehen was in diesem Dokument passiert. Dazu habe ich mir einige, später Folgende, Hilfen zur Hand genommen. Nachdem dieses Nebeneinanderstellen erfolgreich codiert wurde, habe ich mir weiterhin die Hyperlinks vorgenommen, die als Buttons dargestellt werden sollen. Dazu habe ich weiter CSS-Regeln aufgestellt, wie einen Rahmen um die Hyperlinks gezogen. Dies habe ich mit 2 Farben und einen nötigen Abstand zum nächsten Button realisiert. So kommt der Buttoneffekt sehr gut zum Vorschein und die Buttons überlappen sich nicht. Dazu füllte wurden diese umrahmten Hyperlinks noch mit einem dunklen Orange gefüllt. So sah der Hyperlink richtig wie ein Button aus. Mit einem Javascript Mäuse-Event war auch gewährleistet das der Button geklickt werden kann wenn der Mauszeiger keinen Text anklickt sondern nur den Rahmen oder die Füllung, was die Aufgabe Button zum Abschluss brachte. Die Ausrichtung der Graphen und Buttons erforderte alle mir bekannten Verschärfungstags, wie div, span und Tabellentags.

Die wurden von mir skaliert und positioniert so das sie ein angenehmeres Gesamtbild abgeben und als Überschrift zu erkennen sind. Mein Plan war es am rechten Rand des Bildschirmes das Code One Logo mit Adresskopf anzeigen zu lassen, welches „mitwandern“ sollte. Damit meine ich das das Logo immer recht oben am Bildschirm zu sehen sein sollte, auch wenn die Seite gescrollt wird. Dies ist mit der CSS-Regel „position:fixed“ natürlich schnell gemacht. Problem dabei war allerdings das die Buttons auch Platz nach unten benötigten und damit die komplette Seite auch nach unten Scrollbar machten. So überdeckte das Logo beim runterscrollen die Buttons oder die Graphen. Somit musste ich zunächst das Bild

am rechten Rand der Seite(!) absolut positionieren, also eine feste Position geben. Als kleine Zwischenaufgabe wurde mir gesagt, dass ich die Überschriften der Seite sowie der Tunnel und Graphen benennbar machen sollte. Um also die Aufgabe umzusetzen habe ich mir

Letzte Bearbeitung: 13.04.11

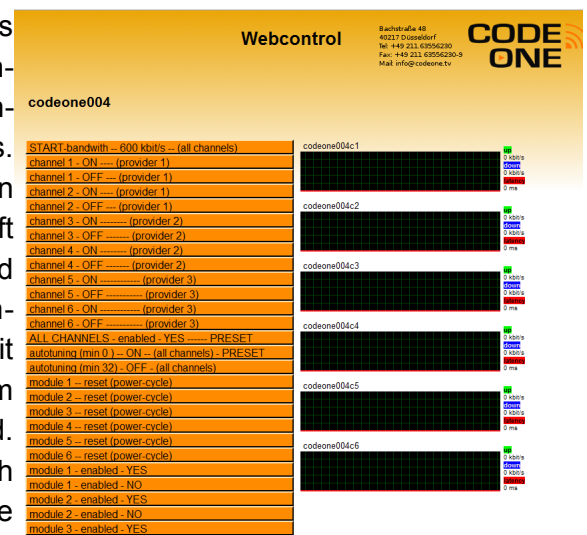


Abbildung 5: Zwischenstand: So sah die Webcontrol nach Abschluss der ersten Aufgabe aus

Autor: David Brüggemann

zunächst überlegt die Eingaben der Überschriften beim Laden der Seite zu abzufragen. Die Umsetzung war auch nicht besonders Zeitintensiv. Dieser erste Ansatz erwies sich aber als zu lästig, da recht viele Abfragen beantwortet werden müssen ohne die Seite vorher gesehen zu haben. Deswegen entschied ich mich die Benennung zunächst wie Vorgegeben anzeigen zu lassen und danach durch einen Klick auf die Überschrift diese umbenennen zu können. Dies realisierte ich indem ich die Überschriften in Hyperlinks umformte, die eine von mir geschriebene Javascript Funktion starten. Die Funktion lässt ein Fenster aufrufen welches eine Eingabe erwartet, die die neue Beschriftung des Textes beinhaltet. Diese Beschriftung wird also nach einem klick auf den „OK“Button der Überschrift zugewiesen. Wird der Vorgang allerdings abgebrochen soll die vorherige Benennung wiederhergestellt werden. Bei dieser Prozedur war es wichtig, dass die Webseite nicht aktualisiert wird, da sonst die Benennungen zurückgesetzt werden. Also ruft der Hyperlink der Überschrift nicht eine neue Seite o.Ä. auf, sondern nur das Event welches die Benennungsfunktion beinhaltet. Schließlich hatte ich damit die erste Aufgabe abgeschlossen. Der Zeitaufwand hierfür belief sich auf ca. 1,5 Wochen.

2.3.2 Zweite Aufgabe

Da die Buttons immer per Hand in die settings.ini Datei eingetragen werden mussten sollte meine nächste Aufgabe sein die Befehle durch ein Dropdown Menü generieren und abschicken zu lassen. Dafür habe ich zunächst die Anordnung vorgesehen, dass das Menü ganz links von der Seite aus zu sehen ist und sich daneben die Graphen wie gewohnt nach rechts ausbreiten. Das Logo, sowie die Überschriften mit ihrer Option zur Benennung sollte dabei erhalten bleiben. Um zu erkennen was generiert und wie es abgeschickt werden sollte sah ich mir zunächst einen Abschiekvorgang mit einem Debugger an. Danach notierte ich welche Befehle abschickbar sein müssten. Dazu sah ich mir die aktuellste settings.ini Datei, welche mir zur Verfügung stand an. Schließlich schaffte ich es das gewünschte Befehlsauswahlmenü zu erstellen. Um jetzt einen Befehl abschicken zu können sollten folgende Schritte absolviert werden:

- Zunächst muss ein Tunnel ausgewählt werden, an welchen der Befehl geschickt werden soll. Die Wahl trifft man aus einem HTML-Auswahlmenü heraus. Sämtliche Tunnelnamen, die angezeigt werden, sind durch eine Javascript Funktion schon in diesem Auswahlmenü eingetragen. Dazu ist zu sagen das der Tunnelname indirekt verändert werden kann, da wie oben schon angesprochen die Überschriften der Tunnel durch die Benennungsfunktion durchaus verändert werden können. Sobald der Tunnel also eine neue Bezeichnung erhält steht diese Bezeichnung auch in dem Auswahlmenü zur Verfügung.
- Hat man eine Wahl getroffen sollte man die IP des Tunnels angeben. Diese muss man leider manuell eintragen, weil ich leider die Nummer nicht per Javascript „abfangen“ wie beispielsweise die Tunnelnamen. Ist die IP allerdings einmal eingetragen, so wird quasi eine Referenz zu dem ausgewählten Tunnel erstellt. Also wechselt man den Tunnel wechselt auch die gespeicherte IP.

- Ist dies abgeschlossen muss man wählen ob ein Channel oder ein Modul beeinflusst werden soll. Um zu erklären was der Unterschied zwischen den beiden ist, muss ich ein wenig Ausholen. Der Router ist über mehrere "Module" an das Internet angeschlossen. Als Modul können verschiedene physikalische Netzwerkschnittstellen (UMTS, Ethernet) zum Einsatz kommen. Über diese Module werden jeweils Daten zu derselben Gegenstelle verschickt. Für den Datenverkehr ist diese Strecke zwischen Router und Gegenstelle als eine einzelne virtuelle Verbindung sichtbar, den erwähnten Tunnel. Der Datenstrom innerhalb des Tunnels wird jedoch auf die Channel aufgeteilt, welche üblicherweise nur eine Teilinformation des Gesamtdatenverkehrs erhalten. Diese Channel können dann jeweils einem bestimmten Modul zugewiesen werden. Wird nun eine Option von dem Benutzer ausgewählt, erscheint ein neues Auswahlmenü. Hier kann man nun explizit angeben an welchen Channel/Modul genau man den Befehl schicken will. Die Optionen sind hierbei lediglich von 1-6 durchnummeriert. Zudem ist die Möglichkeit gegeben, dass die Befehle an alle Channel/Modul abgeschickt werden können.
- Ist diese Auswahl ebenfalls abgeschlossen geht es zum Punkt, was für eine Art Befehl abgeschickt werden soll. Dazu sind bis zu 3 Optionen gleichzeitig auswählbar.
 - Hat man zuvor eine Channelmodifikation gewählt sind die Auswahlmöglichkeiten Autotuning, Latenz und Set ON/OFF.
 - Das Autotuning kann hierbei an und aus gestellt werden und stellt damit die Funktion, ob der Channel sich selbst regelt, an oder aus.
 - Bei der Latenz kann man angeben welche Werte für die Verzögerungszeit optimal wären oder maximal erlaubt sind.
 - Mit der Set-Option kann der jeweilige komplette Channel ganz simpel an oder ausgestellt werden.
 - Wählte man zuvor die Modulooption werden die Auswahlmöglichkeiten Autoconfig, Reset und Set ON/OFF.
 - Autoconfig heißt, wie der Name schon sagt, das das Modul sich automatisch konfigurieren soll oder nicht. Auf was genau dies bedeutet möchte ich jetzt nicht näher eingehen.
 - Der Reset startet das Modul neu.
 - Die Set-Option stoppt und startet das Modul.
- Ist nun der gewünschte Befehl „zusammen geklickt“ muss lediglich noch eine Eingabe des Nicks mit Passwort erfolgen, da der Tunnel passwortgeschützt ist.

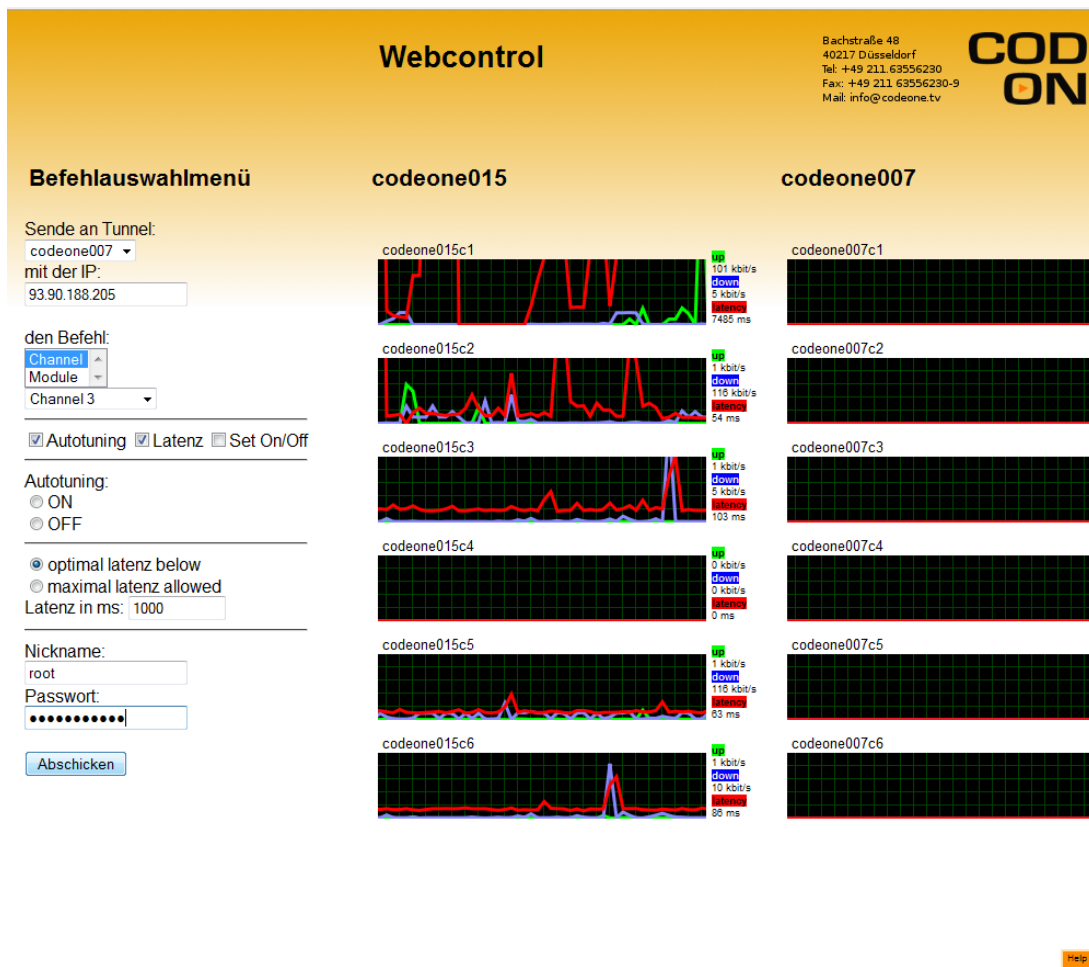


Abbildung 6: Finale Version: Nach der zweiten Aufgabe war meine Bearbeitung so weit komplett. (Seite in Deutsch)

Nun kann der Befehl mit einem Klick auf den Abschieken-Button versendet werden. Vorteil des ganzen: Da die Werte von der Seite erhalten bleiben, können die Befehle an mehrere Channels gleichzeitig versendet werden. Zudem hat man eine viel bessere Übersicht über seine Möglichkeiten und die Tunnel mit den Graphen besser im Blick, weil die Seite nicht mehr so weit (oder gar nicht) sich nach rechts ausbreitet. Zudem ist durch die zusätzlich erforderliche Passwordeingabe mehr Sicherheit für den Benutzer gewährleistet. Zusätzlich zu dem Menü habe ich noch eine Möglichkeit geboten ein eigenes, optionales Benutzerlogo einzufügen im Header links neben der Überschrift der Webseite. Außerdem noch einen Hilfe Button der ein Fenster aufruft welches eine Hilfsseite beinhaltet. Diese erklärt die Funktionsweise und Benutzung der Webcontrol. Da das ganze auch an englische Kunden gehen sollte musste ich die Seite allerdings noch in Englisch übersetzten. Getestet habe ich die Seite nur mit Firefox 3.6 allerdings sollte die Seite mit jeder neueren Firefoxversion korrekt dargestellt werden. Der gesamte Zeitaufwand beläuft sich auf ca. 4,5 Wochen.

2.4 Hilfsmittel

Um das Projekt zu bearbeiten griff ich auf ein paar mir bekannten Hilfsmittel zurückgreifen die ich hier aufführen möchte:

- *HTML* - Die Hypertext Markup Language , ist eine textbasierte Auszeichnungsschrift zur Strukturierung von Inhalten wie Texten, Bildern und Hyperlinks in Dokumenten. HTML-Dokumente sind die Grundlage des World Wide Web und werden von einem Webbrowser dargestellt.
- *CSS* - Eine reine HTML-Seite sieht noch langweilig schwarz-weiß aus. Mit CSS wird die Seite viel angenehmer zu lesen, da der Stylesheet ermöglicht Regeln/Klassen für HTML-Tags aufzustellen die besagen wie sie darzustellen sind.
- *Javascript* - JavaScript beschreibt eine dynamisch typisierte, objektorientierte, aber klassenlose Skriptsprache. Sie wird allen objektorientierten Programmierparadigmen gerecht. In JavaScript lässt sich objektorientiert und sowohl prozedural als auch funktional programmieren. Während HTML/CSS nur wenige, grundlegende Möglichkeiten zur Nutzerinteraktion bietet, können mit JavaScript Inhalte generiert und nachgeladen werden.



Abbildung 7:
Logo wikibooks:
Javascript



Abbildung 8:
offizielles Logo:
Notepad++

- *Notepad++* - ist ein freier Texteditor für Windows und kompatible Betriebssysteme. Neben einfachen Textdateien können auch Quelltexte bearbeitet werden.
- *Firefox* - ist ein freier Webbrowser von Mozilla.
- *GIMP* - ist ein kostenloses und freies Bildbearbeitungsprogramm.
- *Firebug* - ist eine Erweiterung für den Mozilla Firefox Webbrowser und erlaubt die Fehlersuche, die Bearbeitung und das Monitoring von Webseiten. Untersuchen lassen sich insbesondere CSS, HTML, DOM, JavaScript-Elemente einer Webseite. Der Netzverkehr zeigt für alle Elemente einer Webseite die genauen Zeitpunkte der Anfrage und den Zeitraum für das Warten und das Senden der Serverantwort in einer Zeitleiste an. Firebug verfügt zudem über ein JavaScript-Konsole.



Abbildung 9:
offizielles Logo:
GIMP

Links zu den Hilfsmitteln und für das Projekt verwendete Referenzen befinden sich in den Quellen. Alle Verwendeten Programme sind OpenSource und damit kostenlos zu verwenden.

3 Wertung

Um das Projekt anzugehen konnte ich mein Wissen über HTML, CSS und Bildbearbeitung gut einbringen. Zusätzlich musste ich mich intensiv mit Javascript auseinandersetzen, welches mir bis dahin unbekannt war. Allerdings konnte ich dadurch durchaus auch meine Programmier- und auch Recherchierkenntnisse gut einbringen. Das Arbeitsklima bei Code One war sehr angenehm. Ich konnte bei Fragen sich an jeden Mitarbeiter wenden, die sich alle gerne für mich kurz Zeit nahmen um mir qualifizierte Hilfestellung zu geben konnten.

4 Quellen

- <http://codeone.tv>
 - Firmeninterne Mitarbeiter
- <http://notepad-plus-plus.org/>
- <http://www.mozilla-europe.org>
- <https://addons.mozilla.org/de/firefox/addon/firebug/>
- <http://www.gimp.org/>
- <http://de.selfhtml.org/css/>
- <http://de.selfhtml.org/javascript/index.htm>
- <http://de.selfhtml.org/html/index.htm>
- Bericht erstellt mit <http://www.libreoffice.org/> (Libre Office 3.3)

Stand 13.04.2011