

3 MATERIAL UND METHODE

Das "Informationsprogramm zu Mängelansprachen bei der Herstellung von Brühwürsten" entstand als gemeinsames Pilotprojekt der DLG (Deutsche Landwirtschafts-Gesellschaft) und des Instituts für Lebensmittelhygiene der FU-Berlin im Rahmen der vorliegenden Disseration.

Ziel war es, ein interaktives Nachschlagewerk auf CD-ROM zu schaffen, welches auf der Basis der DLG-Prüfbögen vielfältige Hinweise zur Fehlervermeidung in sehr kompakter Form am Beispiel der Brühwürste gibt.

3.1 Programmentwicklung

3.1.1 Zielgruppe

Das Programm soll zwei Zielgruppen erreichen. Einerseits kann es der Lehre an Universitäten, Fachhochschulen, Berufsschulen und handwerklichen Ausbildungszentren dienen. Andererseits wäre auch die Verwendung zur Fehlerbeseitigung im Fleischerhandwerk und der Fleischwarenindustrie eine wesentliche Einsatzmöglichkeit. Dementsprechend heterogen stellt sich auch die Zielgruppe sowohl hinsichtlich des lebensmitteltechnologischen Vor- bzw. Fachwissens als auch der Kenntnisse bezüglich des Umgangs mit Computern dar.

3.1.2 Konzeption

Um eine sinnvolle Konzeption zu finden, müssen Umfang und Inhalte der Anwendung vor Beginn des Programmierens fixiert werden. Nach Definieren der Zielgruppe wurde ein Grundkonzept für die Strukturierung des Programmes entwickelt, welches sich an den unter 3.1.3 aufgeführten Vorgaben orientierte.

Erst nach der Fertigstellung eines Prototypen, der diesem Grundkonzept entsprach, begannen die Literaturrecherchen sowie das Einarbeiten von lebensmitteltechnologischen Inhalten in das Programm.

3.1.3 Zielvorgaben zur Programmentwicklung

- Das Programm soll auf CD-ROM wie auch im Internet lauffähig sein.
- Ein gezielter Zugriff auf Informationen soll schnell und einfach möglich sein.
- Sämtliche enthaltenen Informationen sollen wahlweise auch aus einer Titelgesamtübersicht abrufbar sein.
- Sämtliche Informationen sollen wahlweise auch nacheinander abrufbar sein (vergleichbar mit dem seitenweisen Durchblättern eines Buches).
- Die Benutzerführung soll möglichst einfach und übersichtlich sein.
- Das Programm soll möglichst plattformunabhängig sein.
- Das Layout soll sowohl den weniger versierten als auch den erfahrenen Computernutzer ansprechen.
- Informationen sollen ausgedruckt werden können.
- Die DLG-Prüfbestimmungen sollen in das Programm integriert werden.
- Programmbedienungshinweise sollen integriert werden.

3.2 Auswahl der Computersprache

Ursprünglich war geplant, das Programm so zu erstellen, daß es auf CD-ROM vervielfältigt werden kann. Nachdem das Internet ab 1997 immer populärer wurde und die Zahl der Teilnehmer am Internet stetig zunahm, bot es sich an, das Programm so zu konzipieren, daß es sowohl auf CD-ROM als auch im Internet lauffähig ist. Spätere Aktualisierungen oder Folgeversionen des Programmes würden dann ggf. nur noch im Internet zur Verfügung gestellt, was eine Vervielfältigung auf CD-ROM mit aufwendiger Verteilung an Interessenten völlig überflüssig machen würde.

Des weiteren sollte das Programm möglichst plattformunabhängig sein, d. h. es muß gewährleistet bleiben, daß trotz der inhomogenen Zielgruppe und der damit verbundenen unterschiedlichsten Hardware-Voraussetzungen das Programm auf all diesen verschieden eingerichteten Rechnern lauffähig ist.

Ferner sollte das Programm nicht in einem Format erstellt werden, das eine geringe Verbreitung besitzt oder spezielle Software benötigt und nur nach Erwerb teurer Lizenzen vervielfältigt und abgegeben werden darf.

Auf Grund der dargestellten Überlegungen wurde die Lösung gefunden, das Programm wie Web-Seiten im Internet mit den dort verwendeten Computersprachen *HTML* (Hyper Text Markup Language) und *JavaScript* zu programmieren bzw. zu schreiben. Die oben geschilderten Vorgaben lassen sich damit wie folgt erfüllen:

- 1) Die Plattformunabhängigkeit wird auf Grund der Entwicklung von unterschiedlichen Browser-Versionen weitestgehend gewährleistet. D. h. nicht die Programmierung der Web-Seiten ist letztlich maßgebend, sondern die Funktionalität der verwendeten Browser, um deren Plattformunabhängigkeit sich die Browserhersteller kümmern müssen, da die Verbreitung der Browser ganz entscheidend von ihrer Plattformunabhängigkeit bestimmt wird.
- 2) Die notwendige Software (Browser) ist auf vielen Rechnern schon vorhanden. Zum einen enthält der Lieferumfang von Windows 95 den Microsoft Internet Explorer (MSIE) und zum anderen finden sich in der Zielgruppe auch Internetnutzer, die somit ohnehin einen Browser besitzen. Alles in allem muß zum Betrachten des Programmes meist keine zusätzliche Software installiert werden.
- 3) Die Weitergabe und Vervielfältigung des Microsoft Internet Explorers (einer der am weitesten verbreiteten Browser im Internet) erfordert keine Kosten, was den finanziellen Gesamtaufwand des Projektes in Grenzen hält, da keine teuren Lizenzen für das Vervielfältigen und den Vertrieb von Software erworben werden müssen.
- 4) Für die Programmierung und Erstellung des Programmes wird keine aufwendige Software benötigt, da sich das Einrichten von Web-Seiten mit jedem einfachen Texteditor bewerkstelligen läßt. Dadurch können die Gesamtkosten des Projektes weiter verringert werden.

Die Einführung des World Wide Web (WWW), welches nur ein Teil des Internets darstellt, erzeugte die erste große Popularitätswelle des Internet. Insbesondere führte es dazu, daß sich auf einmal fast jedem die Möglichkeit eröffnete, Dokumente

zu beliebigen Themen per Knopfdruck mehreren Millionen von Internet-Nutzern weltweit zugänglich zu machen (KOCH, 1999).

Die Computersprache, mit der Dokumente für das WWW im Allgemeinen erstellt werden, heißt *HTML*. Diese Computersprache fand nicht zuletzt ihrer leichten Erlernbarkeit wegen sehr schnelle Verbreitung. Nachteile resultieren aus ihrer geringen Flexibilität. Wegen dieser Statik kann der Anwender kaum in irgendeiner Form auf eine Seite Einfluß nehmen; nichts bewegt sich und die Elemente der Seite bleiben konstant. Um diesen Zustand zu ändern, wurden nach und nach verschiedene Möglichkeiten zur Gestaltung sog. dynamischer Web-Seiten entwickelt. Eine dieser so entstandenen Computersprachen stellt *JavaScript* dar, das sich ebenfalls nicht zuletzt seiner verhältnismäßig leichten Erlernbarkeit wegen großer Beliebtheit erfreut (KOCH, 1999).

Für das vorliegende Informationsprogramm wurden die aktuellsten Neuentwicklungen dieser Skriptsprachen (HTML 4.0 und JavaScript 1.3 bzw. JScript 5.0) verwendet, so daß auf diese Weise der zu erwartenden Steigerung der Hard- und Softwarewarenausstattung von Computern während der Programmentwicklungszeit von 2,5 Jahren Rechnung getragen wurde.

3.2.1 HTML

HTML (Hyper Text Markup Language) wurde am Kernforschungszentrum in Genf entwickelt und ist die Computersprache, in der *Web-Seiten* geschrieben werden. Es handelt sich hierbei um eine Auszeichnungssprache, die lediglich markiert, wie bestimmte Textpassagen im nächsten Schritt aussehen sollen. Im Gegensatz zu sog. WYSIWYG-Systemen (What-You-See-Is-What-You-Get), wie z. B. Word, das Texte schon während der Eingabe formatiert, sieht ein Programmierer einer HTML-Seite nicht, wie sich der Text darbietet und aussieht, denn eine Seite mit Auszeichnungsmarkierungen wird erst dann formatiert, wenn sie vom Browser des Lesers dargestellt wird (TOLKSDORF, 1997).

Durch HTML wird dem Computer der allgemeine Aufbau einer WWW-Seite mitgeteilt und durch den Browser läßt sich diese Seite dann visualisieren. HTML beschreibt

dabei nur die generelle Struktur der Seite, die genaue Darstellung bleibt dem Computer in Zusammenarbeit mit dem Browser überlassen (Koch, 1999).

Anders als beispielsweise Word-Dokumente haben HTML-Seiten kein spezielles Dateiformat sondern bestehen nur aus einfachem fortlaufendem Text, ohne Steuerzeichen oder Formatierungsinformationen (sog. ASCII- oder ANSI-Text), was den großen Vorteil bietet, daß diese auf Rechnern mit völlig unterschiedlichen Architekturen verbreitet werden können. Weiterhin lassen sich HTML-Seiten mit jedem beliebigen Text-Editor erstellen (TOLKSDORF, 1997).

3.2.2 JavaScript

JavaScript ermöglicht es als sog. Skriptsprache, schnell und relativ einfach interaktive Web-Seiten zu schaffen. JavaScript wurde im Dezember 1995 von Netscape Communications Corp. entwickelt. Die ursprüngliche Bezeichnung *LiveScript* wurde bald in JavaScript geändert. Die Einbindung von JavaScript in den sehr weit verbreiteten Internet-Browser Netscape Navigator 2.0 trug ihren Teil dazu bei, JavaScript sehr schnell eine große Popularität zu verschaffen. Seit seiner Einführung wurde JavaScript ständig weiterentwickelt und verbessert.

Der größte Konkurrent von Netscape auf dem Browser-Markt war zu dieser Zeit der Internet Explorer von Microsoft. Deshalb war es auch nicht verwunderlich, daß Microsoft eine eigene JavaScript-Version entwickelte. Aus Lizenzgründen durfte aber nicht die Bezeichnung JavaScript verwendet werden, weshalb Microsoft seine Sprache als *JScript* bezeichnete. Prinzipiell ähnelten sich Netscapes JavaScript und Microsofts JScript sehr, jedoch versuch(t)en beide Unternehmen, durch eigene Weiterentwicklung sich von der anderen Firma abzuheben. Bei dieser Entwicklung besteht die Gefahr, daß Web-Programmierer zwei Sprachen erlernen müssen, wenn diese sich nach und nach immer mehr unterscheiden, so daß Web-Seiten letztlich zweimal geschrieben werden müssen, damit sie einer möglichst großen Zahl von Internetnutzern zur Verfügung stehen. Um dieser Entwicklung entgegenzuwirken, wurden im Juni 1997 von der europäischen Organisation ECMA (die ECMA ist ein internationaler, 1961 gegründeter Industrieverband mit Sitz in Genf, der sich der Standardisierung von Informations- und Kommunikations-Systemen widmet) der Standard *ECMAScript* verabschiedet. An diese Norm haben sich nun Netscape und

Microsoft zu halten. Demzufolge sind die Sprachen JavaScript und JScript eigentlich als Implementationen von ECMAScript zu betrachten, was beide Unternehmen aber nicht daran hindert, ihre Sprache mit einer derartigen Geschwindigkeit weiterzuentwickeln, daß die Standardisierung durch die ECMA immer ein Stück hinterherhinkt und de facto die meisten Internetseiten in zwei verschiedenen Versionen programmiert werden müssen.

Obwohl die Sprache offiziell ECMAScript heißt, wird sie im Allgemeinen weiterhin als JavaScript bezeichnet, wobei man darunter sowohl JavaScript von Netscape als auch JScript von Microsoft versteht (KOCH, 1999).

3.2.3 Die Funktion eines Browsers

Vor einigen Jahren bestand das Internet überwiegend aus Newsgroups, Chats und Download-Bibliotheken. Durch die Etablierung des WWW als Standard im Internet wurden vermehrt redaktionelle Seiten (Web-Seiten, Web-Pages, Internet-Seiten) angeboten. Um die Inhalte dieser Seiten darstellen zu können, benötigt man eine spezielle Software, die als Browser (wörtlich übersetzt "Durchsucher") bezeichnet wird. Nach Eingabe einer bestimmten Internetadresse stellt der Browser eine Verbindung zu dieser Seite im Internet her und zeigt die dort vorhandenen Inhalte an. Dieses einfache Prinzip funktioniert aber nicht nur im Internet, sondern genauso lokal auf einem Rechner. Anstelle des Internets werden dort Inhalte, die in den gleichen Computersprachen wie die Seiten im Internet verfaßt wurden, vom Browser von der Festplatte oder von einer CD-ROM geladen und angezeigt. Damit läßt sich die Vorgabe dieser Arbeit, ein Programm sowohl auf CD-ROM als auch im Internet darstellen zu können, sehr gut realisieren.

Die Browserhersteller sind bemüht, ihre Produkte ständig weiter zu entwickeln, um sich durch das Einarbeiten diverser Neuerungen einem noch größeren Kundenkreis zu erschließen. Erfahrungsgemäß dauert es eine gewisse Zeit, bis die neueren Versionen von den Internetnutzern angenommen werden. Die Browsermarktanteile verhalten sich daher unter Berücksichtigung der unterschiedlichen Versionen Anfang 2000 wie folgt:

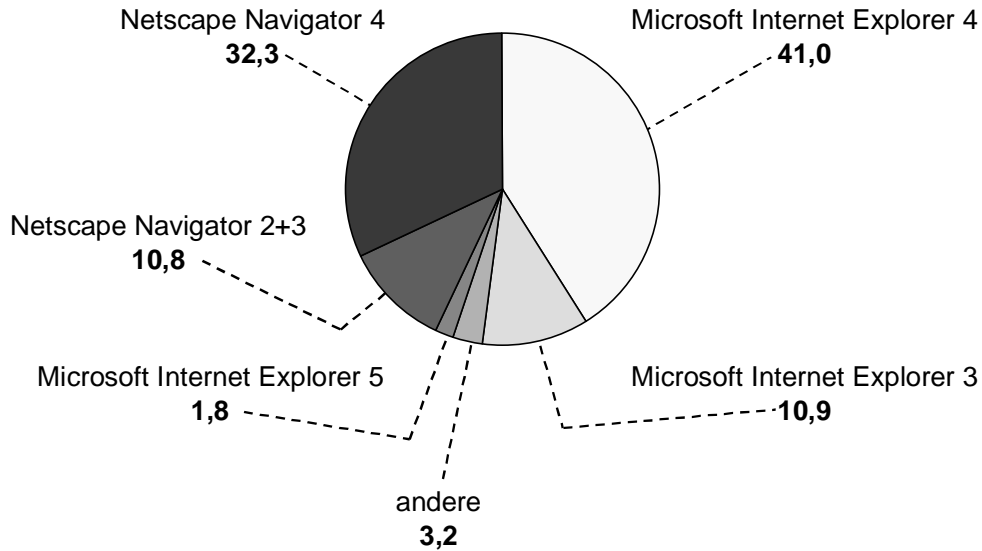


Abb. 3.1 : Marktanteile verschiedener Browser in Prozent (DE THIER, 2000)

Aus Abbildung 3.1 wird ersichtlich, daß außer den beiden Marktführern Microsoft Internet Explorer (53,7 % Marktanteil) und Netscape Navigator (43,1 % Marktanteil) andere Browser (3,2 % Marktanteil) nur eine sehr untergeordnete Rolle spielen. Daher wurde bei der Programmierung darauf geachtet, daß Kompatibilität mit dem Microsoft Internet Explorer und dem Netscape Navigator besteht.

3.3 Systemanforderungen, Hard- und Softwarevoraussetzungen

Um den Grundgedanken zu realisieren, das "virtuelle Lehrbuch" sowohl auf CD-ROM als auch im Internet benutzen zu können, wurde entschieden, das Programm in der Dokumentenbeschreibungssprache HTML und der Skriptsprache JavaScript zu programmieren. Folglich werden die zur Betrachtung des Programmes notwendigen Systemvoraussetzungen von den hierfür verwendeten Browsern bestimmt.

Da der Microsoft Internet Explorer schon zu Beginn des Vorhabens kostenlos vervielfältigt und weiterverbreitet werden durfte, richteten sich die Systemanforderungen für die CD in erster Linie nach den Systemanforderungen für diesen Browser.

Als Zielplattform wurde daher ein IBM-kompatibler Personal-Computer vorausgesetzt. Dieser sollte mindestens einen 486er-Prozessor und 16 Megabyte Arbeitsspeicher besitzen. Ferner wurden und werden ein CD-ROM-Laufwerk, eine

Grafikkarte mit mindestens 65.000 Farben und eine minimale Bildschirmauflösung von 800 x 600 Pixeln benötigt.

3.4 Generelle Schwierigkeiten bei der Programmerstellung

3.4.1 Unterschiedliche Darstellungsmodi

Ein nicht modifiziertes Browserfenster besteht im Allgemeinen aus den in Abb. 3.2 dargestellten Navigationskomponenten.

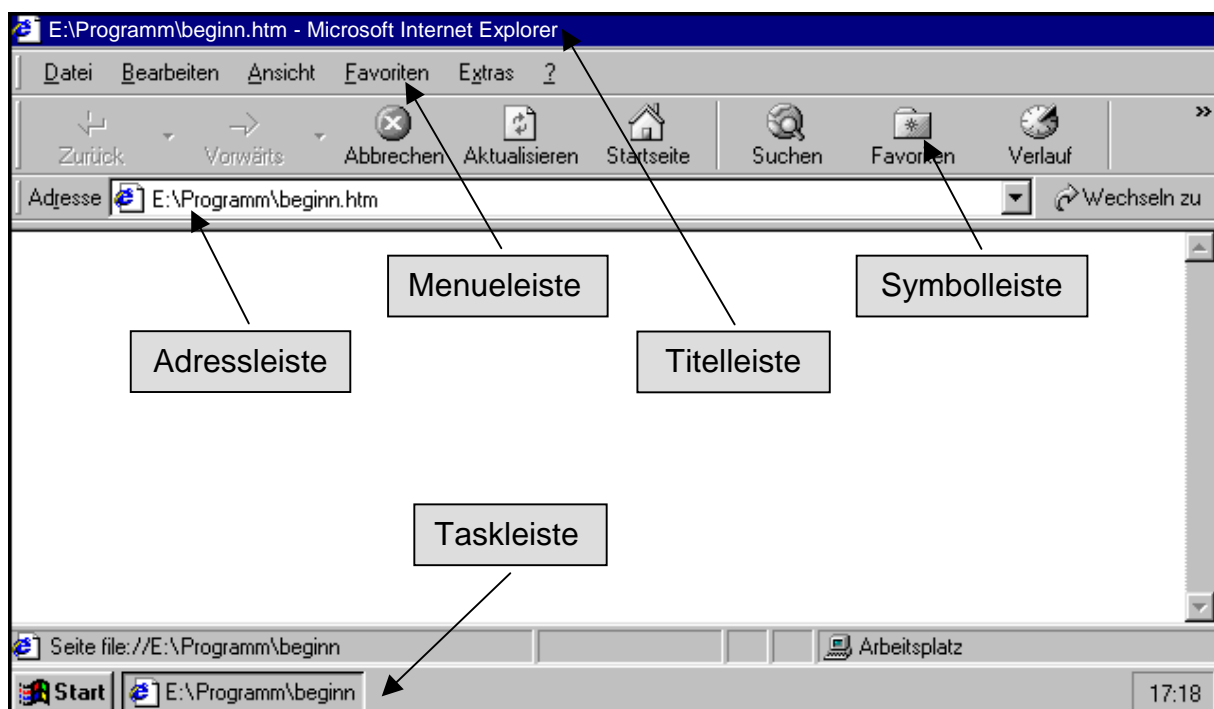


Abb. 3.2 : Verschiedene in einem Browserfenster vorhandene Navigationskomponenten am Beispiel des Microsoft Internet Explorers

Je nach Browserhersteller und Browserversion lassen sich diese Komponenten ganz oder teilweise ausblenden sowie ihre genaue Anordnung und Größe unterschiedlich konfigurieren, weshalb der für Darstellungszwecke zur Verfügung stehende Raum dementsprechend verschieden groß ausfallen kann. Diese Navigationskomponenten werden auf Grund ihrer Funktionen überwiegend im Internet (Online) benötigt (z. B. "Aktualisieren", "Suchen", "Favoriten", "Verlauf" usw.). Im Falle eines in sich geschlossenen Programmes, welches lokal (Offline) auf einem Rechner läuft, sind die meisten dieser Funktionen überflüssig bzw. wegen des Offline-

Modus ohnehin bedeutungslos. Des weiteren erweist es sich innerhalb eines Programmes für eine übersichtliche und verständliche Benutzerführung als vorteilhaft, alle benötigten Navigationskomponenten in die Programmoberfläche zu integrieren.

Abb. 3.3 zeigt die Darstellung der Startseite im Normalmodus, d. h. ohne Veränderung irgendwelcher Komponenten am Browserfenster.



Abb. 3.3 : Darstellung der Startseite mit dem Microsoft Internet Explorer im "Normalmodus"

Es ist deutlich zu erkennen, daß die Startseite keinen optimalen Eindruck vermittelt. Weil die eingeblendeten, aber nicht benötigten Browserkomponenten sehr viel Raum einnehmen, muß man scrollen, um die ganze Startseite betrachten zu können, wozu ein entsprechender Scrollbalken am rechten Bildrand zusätzlich eingeblendet wird.

Als alternative Lösung wurde das Programm so konzipiert, daß diese Browserkomponenten von vornherein völlig ausgeblendet werden, d. h. das Programm läuft im sog. **Vollbild-Modus**. Beim Microsoft Internet Explorer werden dann sämtliche sichtbaren Browserkomponenten ausgeblendet und der gesamte

Bildschirminhalt zu Darstellungszwecken zur Verfügung gestellt, während beim Netscape Navigator die Titelleiste auch im Vollbildmodus erhalten bleibt.



Abb. 3.4 : Darstellung der Startseite mit dem Microsoft Internet Explorer im "Vollbild-Modus"

Wie Abb. 3.4 verdeutlicht, steht im Gegensatz zur vorherigen Abbildung das gesamte Browserfenster für Darstellungszwecke zur Verfügung; es muß nicht mehr gescrollt werden, weshalb auch kein Scrollbalken eingeblendet zu werden braucht.

3.4.2 Unterschiedliche Bildschirm- bzw. Monitorauflösungen

Die Bildschirmauflösung gibt an, aus wie vielen einzelnen Bildpunkten sich das am Monitor dargestellte Bild aufbaut. Im Jahr 1997, als die Entwicklung des Programmes begann, waren 15-Zoll-Monitore (Bildschirmdiagonale von 37,5 cm) die am häufigsten in Verbindung mit einem PC verwendeten Monitore, inzwischen werden die PC häufiger mit einem 17-Zoll-Monitor ausgestattet. Bei einem Monitor

mit einer Bildschirmdiagonalen von 15 Zoll gilt eine Bildschirmauflösung von 800 x 600 Pixeln (Einzelpunkten) als Standard. Das dargestellte Bild ist aus 800 Einzelpunkten in der Breite und 600 Einzelpunkten in der Höhe aufgebaut. Bei einer stärkeren Auflösung (z. B. 1024 x 768 Pixel) ist jeder Einzelpunkt zwar kleiner, das Gesamtbild dadurch aber schärfer, umgekehrt fällt bei einer geringeren Auflösung (640 x 480) der einzelne Bildpunkt größer, das Gesamtbild aber unschärfer aus.

Eine Grafik stellt eine feste Größe in Bezug auf die Anzahl der Bildpunkte dar und wird deshalb bei differierenden Auflösungen unterschiedlich groß dargestellt.

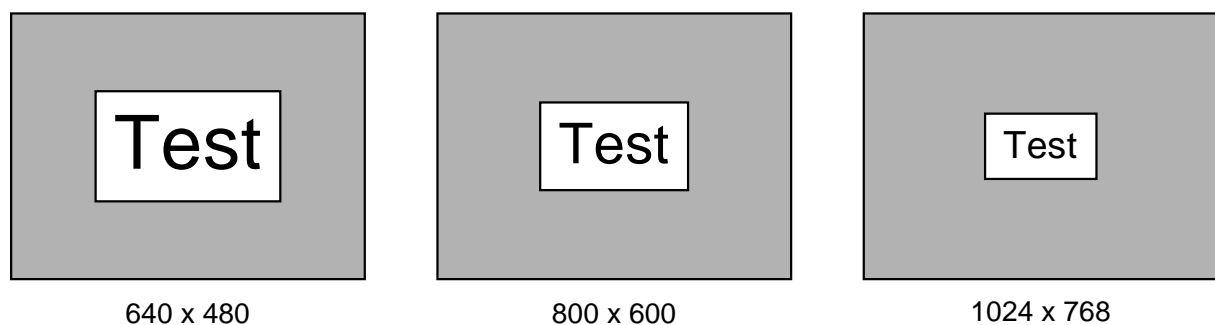


Abb. 3.5 : Darstellung einer Grafik bei verschiedenen Bildschirmauflösungen

Abb. 3.5 verdeutlicht diesen Sachverhalt. So bleibt die Grafik dabei immer die gleiche, während sich die Darstellungsgröße mit zunehmender Auflösung verkleinert und die Bildschärfe umgekehrt proportional zunimmt.

Im Zuge der Erstellung des Programmes bestand das Problem, daß sich bei einer Auflösung von 640 x 480 Pixeln (der Standard-Auflösung bei älteren, manchmal noch verwendeten 14-Zoll-Monitoren) eine Buttonleiste am unteren Bildrand aus einzelnen Buttons mit Beschriftung zwar installieren läßt, diese Beschriftung aber bei der nächsthöheren Auflösung von 800 x 600 Pixeln unlesbar wird, da jeder einzelne Button schon zu klein wiedergegeben wird. Konzipiert man hingegen die Buttonleiste bezüglich der Größe der einzelnen Buttons derart, daß sie sich bei einer Auflösung von 800 x 600 Pixeln ideal darstellt, so wird bei einer Auflösung von 640 x 480 die Buttonleiste zu groß, um gleichzeitig alle Buttons am Monitor in einer Reihe anzeigen zu können. Im Zweifelsfall hätten für unterschiedliche Auflösungen unterschiedlich große Grafiken verwendet werden müssen, was bei der Programmerstellung einen immensen Mehraufwand bedeutet hätte. Aus diesem Grund

wurde das Programm so angelegt, daß die kleinstmögliche Auflösung 800 x 600 Pixel beträgt, zumal die Entwicklung ohnehin zu größeren Monitoren tendiert.

Beim Programmstart findet eine mit JavaScript programmierte Abfrage statt, bei der



Abb. 3.6 : Startseite bei einer Auflösung von 640 x 480

die momentan verwendete Bildschirmhöhe und -breite festgestellt wird. Beträgt die Auflösung 640 x 480, werden dabei vom System kleinere Werte als 800 x 600 zurückgegeben. Dieser Umstand hat zur Folge, daß eine Fehlermeldung mit der Anforderung an den Benutzer erscheint, die Einstellungen der Grafikkarte dahingehend zu modifizieren, daß die Auflösung mindestens 800 x 600 Pixel beträgt (vgl. Abb. 3.6). Eine Änderung dieser Einstellungen aus dem Programm heraus ist programmieretechnisch nicht möglich, weshalb der Benutzer derartige Anpassungen immer selbst vornehmen muß.

Um das Problem der unterschiedlich großen Darstellung bei verschiedenen



Abb. 3.7 : Startseite bei einer Auflösung von 800 x 600

gegen eine höhere Auflösung (egal wie hoch) vor, so entsteht ein

die momentan verwendete Bildschirmhöhe und -breite festgestellt wird. Beträgt die Auflösung 640 x 480, werden dabei vom System kleinere Werte als 800 x 600 zurückgegeben. Dieser Umstand hat zur Folge, daß eine Fehlermeldung mit der Anforderung an den Benutzer erscheint, die Einstellungen der Grafikkarte dahingehend

Auflösungen zu eliminieren, wurden die einzelnen Seiten so programmiert, daß der eigentliche Programminhalt immer in einer Größe von 800 x 600 Pixeln angezeigt wird. Arbeitet man mit einer Bildschirmauflösung von 800 x 600 Pixeln, nimmt das Programm demgemäß den gesamten Bildschirm ein (vgl. Abb. 3.7). Liegt hin-



Abb. 3.8 : Startseite bei einer Auflösung von 1024 x 768

Rahmen (als hellgrauer Hintergrund) um den gezeigten Programmausschnitt. Der Rahmen ist in seiner Größe flexibel und paßt sich immer der verwendeten Auflösung an, so daß der dargestellte Programmausschnitt innerhalb des Rahmens stets 800 x 600 Pixel umfaßt (vgl. Abb. 3.8). Da unterschiedliche Auflösungs-

en nicht mehr berücksichtigt werden müssen, sind die einzelnen Seiten viel einfacher zu programmieren.

3.5 Ursprüngliche Planung der Programmstruktur

Bei der ursprünglichen Konzeption wurde folgende Grobstruktur des Programmes in Papierform erstellt (s. Abb. 3.9).

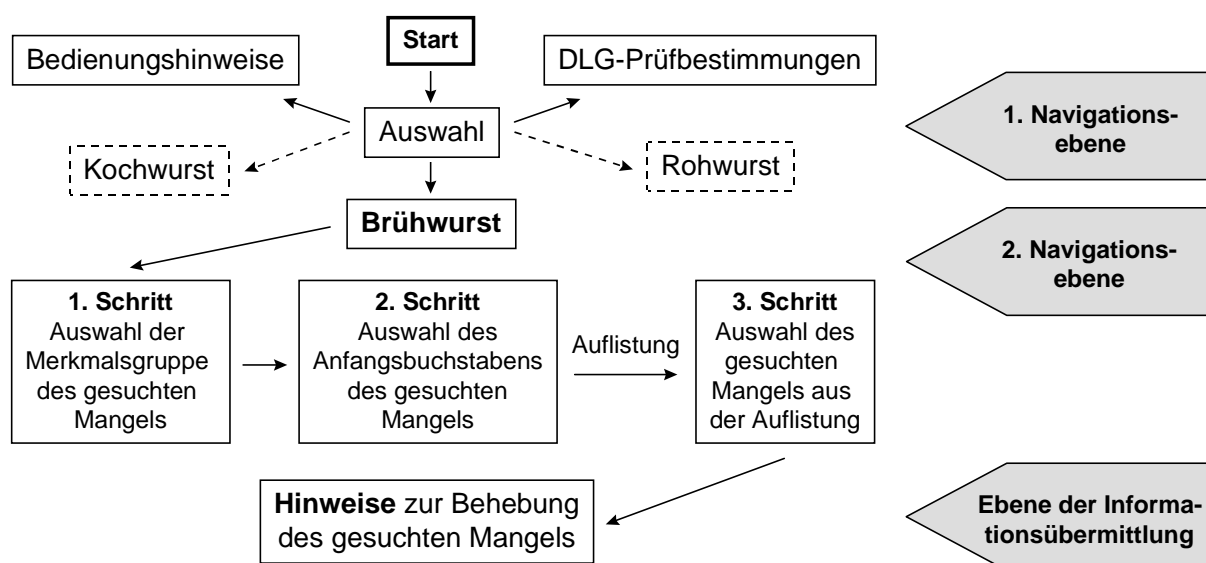


Abb. 3.9 : Ursprüngliche Planung der Programmstruktur

3.5.1 Die erste Navigationsebene

Nach dem Öffnen der Startseite des Programmes sollte der Benutzer auf eine Seite gelangen, die ihm die Auswahl unter den verschiedenen Wurstkategorien (Brühwurst, Kochwurst, Rohwurst), den Programmbedienungshinweisen und den Prüfbestimmungen der DLG ermöglicht.

Verschiedene Wurstkategorien

Am Anfang des Projektes war vorgesehen, auch die Kategorien Rohwurst und Kochwurst mit in das Programm einzuarbeiten. Auf Grund der teilweise sehr komplexen Materie und dem damit verbundenen Zeitaufwand, wurde die Planung alsbald dahingehend modifiziert, daß der Umfang des Programmes vorerst auf die Kategorie der Brühwürste als technologisch relativ homogene Produktgruppe beschränkt bleiben sollte.

Bedienungshinweise

Vor dem eigentlichen Programmeinstieg sollte es möglich sein, sich über den Umgang mit dem Programm eingehend zu informieren.

DLG-Prüfbestimmungen

Die DLG-Prüfbestimmungen informieren über den Ablauf der DLG-Qualitätswettbewerbe und die dafür notwendigen Rahmenbedingungen. Von Anfang an war geplant, diese mit in das Programm einzuarbeiten.

3.5.2 Die zweite Navigationsebene

Eine Eingabemöglichkeit für Suchbegriffe mit anschließender Suchfunktion ist unter alleiniger Verwendung von HTML nicht möglich, hierfür werden zusätzlich spezielle Computersprachen wie JavaScript benötigt. Zum damaligen Zeitpunkt (1996) wurde "JavaScript" (Version 1.0) von einem der gängigsten Browser (Microsoft Internet

Explorer 2.0) noch nicht einmal unterstützt. Darüber hinaus war die Entwicklung der Programmiersprache "JavaScript" noch nicht sehr weit fortgeschritten, weshalb sich eine entsprechende Programmierung auch für die neueren Browserversionen (Microsoft Internet Explorer 4.0 und Netscape Navigator 4.0) noch nicht absehen ließ. Als Ausweg sollte die gezielte Informationssuche derart gestaltet werden, daß der Programmbenutzer erst die Merkmalsgruppe (siehe Abb. 3.10 rechts) und nachfolgend den Anfangsbuchstaben des gesuchten Mangels (siehe Abb. 3.10 links) anklicken muß, um dann eine Auflistung der hierzu vorhandenen Titel zu erhalten (siehe Abb. 3.10 Mitte). Nach erfolgter Auflistung und dem Anklicken des gesuchten Titels wurden die entsprechenden Informationen angezeigt (**Ebene der Informationsübermittlung**).

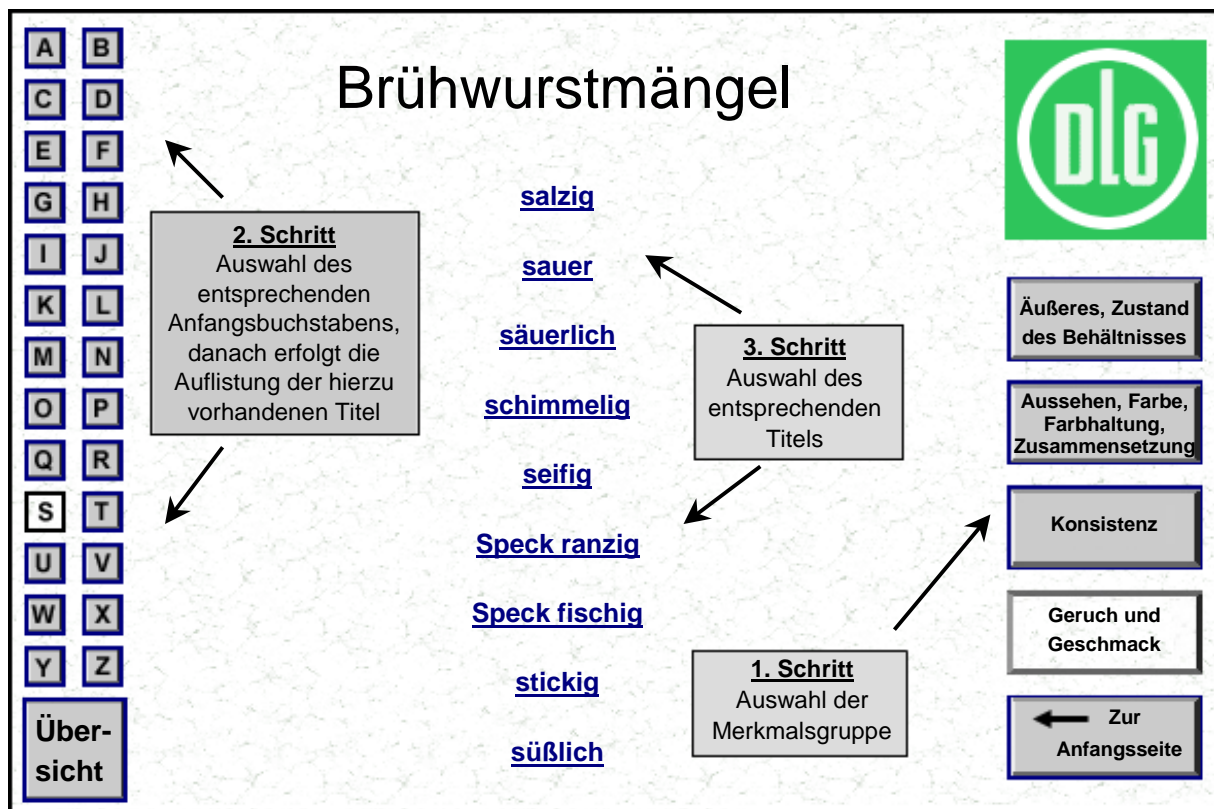


Abb. 3.10 : Ansicht der 2. Navigationsebene

3.6 Strukturierung der Zugriffsmöglichkeiten nach Weiterentwicklung von JavaScript und dem Erscheinen neuer Browsergenerationen

Recht bald wurde klar, daß die oben beschriebene Form der Informations-

bereitstellung über die Auswahl der Merkmalsgruppe und dem nachfolgenden Anklicken des betreffenden Anfangsbuchstabens eine sehr umständliche Lösung bildete. Am Jahresanfang 1998 kamen dann der Netscape Navigator 4.0 und der Microsoft Internet Explorer 4.0 auf den Markt. In beiden Browsern implementierten die Entwickler Neuerungen von JavaScript, so daß ab diesem Zeitpunkt die Möglichkeit bestand, eine Eingabemaske für Suchbegriffe zu erstellen und das Suchergebnis entsprechend aufzulisten.

Als Folge änderte sich die Programmstruktur grundlegend, indem sich eine Einteilung in unterschiedliche Navigationsebenen erübrigte. Stattdessen befinden sich nun auf jeder Seite des Programmes alle allgemeinen Navigationsmöglichkeiten, der Seite entsprechend angepaßt, in einer "Buttonleiste" am unteren Bildrand. Weitere seitenspezifische Navigationskomponenten sind teilweise in den auf der Seite angezeigten Text eingepaßt.

3.6.1 Navigationskomponenten der Buttonleiste

3.6.1.1 Suchseite

Auf allen Seiten, mit Ausnahme der Suchseite, befindet sich ein Button, um auf die Suchseite zu gelangen. Von hier aus ist es möglich, sämtliche im Programm enthaltenen Informationen abzurufen. Die Struktur der dafür notwendigen Navigationsmöglichkeiten findet sich in Abb. 3.11.

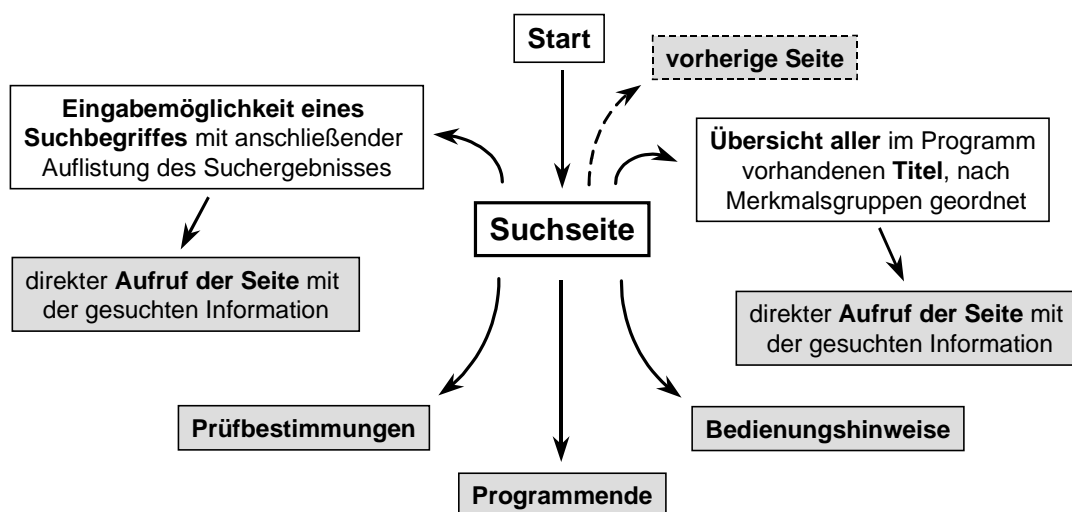


Abb. 3.11 : Von der Suchseite ausgehende Navigationsmöglichkeiten

3.6.1.2 Programmende

Auf allen Seiten des Programmes, mit Ausnahme der Startseite, findet sich ein Button, um das Programm zu beenden bzw. den Browser zu schließen. Beim **Netscape Navigator** schließt sich nach Betätigung des Buttons der Browser sofort, während beim **Microsoft Internet Explorer** auf ein Fenster gelinkt wird, bei dem vor dem Schließen nochmals ein Popup-Fenster erscheint, in dem der Benutzer gefragt wird, ob er das letzte geöffnete Fenster (und damit den Browser) auch wirklich schließen möchte. Erst nach Anklicken einer Bestätigung schließt sich auch dieses letzte Fenster. Diese Lösung resultiert aus der Sicherheitsstrategie von Microsoft, die ein selbständiges Schließen des ersten geöffneten Fensters ohne Rückfrage nicht erlaubt.

3.6.1.3 Bedienungshinweise

Ein weiterer Button auf allen Seiten des Programmes, mit Ausnahme der Startseite, führt zu den Bedienungshinweisen. Diese enthalten Informationen zum Umgang mit dem Programm oder zu bestimmten empfohlenen Einstellungen in der Systemsteuerung, welche notwendig sind, um ein einwandfreies Funktionieren des Programmes zu gewährleisten.

3.6.1.4 Prüfbestimmungen

Alle Seiten des Programmes, mit Ausnahme der Startseite, zeigen einen Button, der zu den DLG-Prüfbestimmungen führt. Dabei wird eine Auswahlseite geöffnet, die eine Gesamtübersicht mit Zugangsmöglichkeit zu allen 14 Kapiteln dieser Bestimmungen enthält.

3.6.1.5 Drucken

Übersichtseiten, die lediglich der Navigation dienen, enthalten in ihrer Buttonleiste keinen Druckbutton. Andere Seiten im Programm, deren Inhalt auszudrucken sinnvoll sein kann, besitzen einen "Druckbutton". Dieser öffnet das Dialogfenster,

welches ebenso beim Erteilen eines Druckauftrages aus der Navigationsleiste des betreffenden Browser heraus erscheinen würde und ermöglicht den Ausdruck der geöffneten Seite.

3.6.1.6 Vorherige Seite

Auf allen Seiten, mit Ausnahme der Startseite, befindet sich ein Button, der den Zugang auf die jeweils unmittelbar zuvor aufgerufene Seite ermöglicht, d. h., die zuletzt gezeigten Seiten lassen sich mit dieser Funktion in chronologischer Reihenfolge rückwärts abarbeiten.

3.6.1.7 Nächste Seite

Einen weiteren Button enthält die Navigationsleiste von Seiten, die nur einen Teil eines zusammengehörenden größeren Teiles darstellen. Dazu gehören die DLG-Prüfbestimmungen mit ihren 14 Kapiteln und die Informationen zur Mängelansprache, die der Reihenfolge des DLG-Prüfschemas für Brühwürste entsprechend von 1-156 durchnummeriert sind.

3.6.2 Nicht in der Buttonleiste vorhandene Navigationskomponenten

3.6.2.1 Suchen

Ein Button mit Suchfunktion befindet sich auf der Suchseite. Nach Eingabe eines Suchbegriffes und Starten der Suche wird eine Liste sämtlicher im Programm vorhandenen Titel nach diesem Begriff überprüft. Nachfolgend werden diejenigen Titel, die den Suchbegriff enthalten, links auf dieser Seite aufgelistet. Durch Anklicken des gesuchten Titels läßt sich die entsprechende Seite aufrufen.

3.6.2.2 Seitenübersicht

Ein Button zur Seitenübersicht befindet sich ebenfalls auf der Suchseite. Nach dem

Starten der Seitenübersicht werden alle im Programm vorhandenen Titel, nach Prüfmerkmalsgruppen geordnet, aufgezeigt. Durch Anklicken des gesuchten Titels läßt sich die entsprechende Seite aufrufen.

3.6.2.3 Bildansicht

Am Anfang von Seiten, die Informationen zu Mängeln enthalten und welche eine grafische Darstellung eines mit diesem Mangel behafteten Brühwurstproduktes im Programm beinhalten, befindet sich ein Button, der die entsprechend Grafik anstelle des zuvor gezeigten Textes aufruft.

3.6.2.4 Einzelne Kapitel der DLG-Prüfbestimmungen

Auf der Auswahlseite der DLG-Prüfbestimmungen gibt es für jedes Kapitel einen beschrifteten Button, durch den das entsprechende Kapitel aufgerufen werden kann.

3.6.2.5 Grüne Links

Inhaltlich zusammenhängende Seiten sollten miteinander verbunden werden. Diese Verweise erfolgen mit Hilfe von Links, welche auf die in dem grün hervorgehobenen Text genannte Seite führen.

3.6.2.6 Blaue Links

Blaue Links führen auf eine Seite, die Erläuterungen oder Definitionen des im Text blau hervorgehobenen Begriffes enthält. Hierdurch sollte dem technologisch weniger geschulten Nutzer ein Basiswissen in Art eines "Lexikons der Nomenklatur" an die Hand gegeben werden.

3.7 Verwendete Medien

3.7.1 Texte

Die fachlichen Texte zur Behebung oder Vermeidung der einzelnen Fehler wurden in

Anlehnung an die gängigsten Lehrbücher sowie zahlreiche wissenschaftliche Veröffentlichungen (siehe Schriftum) erstellt.

Die dabei verwendete Sprache soll einfach sein, Akronyme, Abkürzungen und Fachsprache sind möglichst zu vermeiden (STEENS, 1998).

In dem vorliegenden Programm wurde auf die Verwendung von kurzen Sätzen und den weitestgehenden Verzicht von Nebensätzen besonderer Wert gelegt. Da lebensmitteltechnologische Fachbegriffe einen Teil des zu vermittelnden Wissens darstellen, wurde auf die Verwendung solcher Begriffe nicht verzichtet, zumal diese stets als Link (vgl. 3.6.2.6) dargestellt und damit entsprechend erklärt werden.

Der Inhalt der DLG-Prüfbestimmungen wurde wörtlich den Ausführungen der „DLG-Prüfbestimmungen für Fleischerzeugnisse, Fertiggerichte, Tiefkühlkost und Feinkost“ entnommen.

3.7.2 Bildmaterialien

Die fachlichen Abbildungen, die während zahlreicher Qualitätswettbewerbe für Brühwürste entstanden sind, wurden von der DLG in Form von Dias und Photographien für das vorliegende Programm zur Verfügung gestellt. Dias wurden nach Einsendung an ein Photolabor dort eingescannt und auf CD-ROM gebrannt, Photographien wurden selbst eingescannt. Die weitere digitale Bearbeitung erfolgte mit dem Programm Adobe Photoshop 4.0[®]. Qualitative Mängel ließen sich durch Korrektur von Tonwerten, Helligkeit, Kontrast und Schärfe sowie durch zahlreiche Filterfunktionen des Programmes beheben. Aus didaktischen Gründen wurden manche Bildelemente hervorgehoben oder vervielfacht, wohingegen störende Bildelemente grundsätzlich entfernt wurden.

3.8 Verwendete Hard- und Software

3.8.1 Software

Folgende Software wurde zur Programmerstellung verwendet:

| Name | Hersteller | Verwendung |
|--------------------------------------------------------------------------------------------------------------------|-------------------------------------|----------------------------------------------------------|
| Adobe Photoshop 4.0 [®] | Adobe Systems Incorporated | für die digitale Bildbearbeitung |
| CeQuadrat WinOnCD [®] | CeQuadrat [®] GmbH | zum Brennen von CD's |
| Luckman Interactive's Web Edit Pro [®] | Luckman Interactive Incorporated | Editor zum Erstellen des HTML und JavaScript-Quelltextes |
| Microsoft Internet Explorer [®] 4.0 Microsoft Internet Explorer [®] 5.0 | Microsoft Corporation | als Browser |
| Microsoft Internet Explorer Administration Kit [®] | Microsoft Corporation | zum Einrichten des MSIE auf CD-ROM |
| Microsoft Windows 95 [®] | Microsoft Corporation | als Betriebssystem |
| Microsoft Word für Windows 95 [®] | Microsoft Corporation | zum Erstellen von Texten |
| Netscape [®] Navigator (bzw. Communicator) 4.0 Netscape [®] Navigator (bzw. Communicator) 5.0 | Netscape Communications Corporation | als Browser |

Tab. 3.1 : Zur Programmerstellung verwendete Software

3.8.2 Hardware

Das Programm wurde auf einem IBM-kompatiblen Personal Computer mit Intel Pentium Prozessor[®], 32 Megabyte Arbeitsspeicher, 133 Megahertz Taktfrequenz, einer Festplatte mit 2,5 Gigabyte und einer Matrix Millenium[®] Grafikkarte erstellt.

Ferner wurden für die Herstellung von Sicherungskopien des Programmes auf CD-ROM ein externer 4 x CD-R-Brenner mit einer 4 Gigabyte-SCSI-Festplatte verwendet.

Verschiedene Abbildungen wurden mit einem hochauflösenden Farbscanner Umax Astra 1200 S[®] digitalisiert.