
Module „Grundlagen und Dynamische Webprogrammierung“

Übergreifende Projektaufgabe mit HTML, CSS, JS und PHP

Prof. Rolf Kruse/Kristof Friess

Projektaufgabe

Thema

Kleine Website für einen realen oder fiktiven Kunden, z.B.

- ❖ ein Geschäft wie Bäcker, Metzger, Fahrradgeschäft, Bioladen, Fotoladen,..
- ❖ einen Dienstleister wie Arztpraxis, Schlüsseldienst, Reparaturwerkstatt, Architekturbüro..
- ❖ einen Event: Konferenz, Jubiläumsfeier,...
- ❖ eine soziale Einrichtung, Verein, Kindergarten,...

Basis für die Auswahl können Ihre Überlegungen aus dem Modul "Datenbanken" sein, so dass Sie auf der dort entwickelten **Datenbank** aufbauen können. Sie können aber auch ein neues Thema wählen und eine passende Datenbank konzipieren und anlegen.

Projektaufgabe

Umfang

- ❖ mind. **6 Hauptseiten**
- ❖ insges. mind. **3 Unterseiten**
- ❖ Seite über **einheitliche Navigation** (mit 2 Ebenen) miteinander verknüpft

Mögliche Themenbereiche

- ❖ Wir über uns, Abteilungen/Unterbereiche, "Zur Geschichte",...
 - ❖ Aktuelles: News, Termine,..
 - ❖ Infoseite, Hintergründe, "Was ist eigentlich..."
 - ❖ Einzelne Produkte/Dienstleistungen, Mitarbeiter, gegleiert nach Kategorien/Mannschaften/Spieltag/Aktualität
 - ❖ Referenzen, Projektbeispiele, unsere Partner,...
 - ❖ Kontakt, Serviceanfragen
-
- ❖ Außerdem ist ein **Impressum** (mit Angaben zu Urhebern und Quellen) einzubinden.

Projektaufgabe

Inhalte, Design und Funktionalität

- ❖ **Inhalte** (Bilder, Überschriften, einführende Texte) und **Design** sollten gut **zum gewählten Thema passen** und modern wirken
 - Holen Sie sich Anregungen bei vergleichbaren, gut gestalteten Webseiten.

- ❖ Die **Aufmachung** mit Texten, Bildern, Grafiken und Tabellen sollte **abwechslungsreich** sein.
 - Fließtext und sich wiederholende Elemente können als Blindtext, Dummy-Grafiken u.ä. ausgeführt sein, solange die prinzipielle Intention erkennbar ist.

- ❖ Unterstützen Sie durch weitere Angaben im Quelltext (u.a. per *title*-, *alt*-, *label*-, *legend*-, *noscript*-Tag) die **Barrierefreiheit** und sichere Nutzbarkeit der Seiten.

- ❖ **Interaktivität**: Mit Hilfe von Stilvorlagen und Javascript, müssen - außer der Seitennavigation - an mind. einer Stelle auch weitere Seitenelemente auf den Nutzer reagieren
 - z.B. bei Mouse-Over oder Erscheinen/Ausklappen von Bereichen durch ein Schaltfeld

Projektaufgabe

Technologien und Formate 1

Valides HTML5

- ❖ **Logische Strukturierung** mit *div, nav, header, article* etc.
- ❖ **Bilder** werden in einem Verzeichnis auf dem Server gespeichert und relativ verlinkt
- ❖ Verwenden Sie für **Formulare Gliederungs- und Hinweiselemente** wie *fieldset, label,...*

CSS3

- ❖ **Transitions**: Die Darstellung einiger Elemente wird bei Nutzerinteraktionen verändert
- ❖ Auswahl und lokale Einbindung mind. eines **Webfonts**
 - Schriftarten z.B. von Google Webfonts <https://www.google.com/fonts#>
- ❖ **Responsive Design**: Automatische Anpassung des Layouts für mind. 2 Bildschirmgrößen (*Small* und *Medium*) per „*Media Queries*“
 - Umfluss von Bildern/Texten mit `<float>` oder altern. Methoden (`inline-block`, `flexbox`,...)
 - Tipp: Entwicklung Sie zuerst für *Small*, dann für *Medium* („Mobile First“)
- ❖ **Animation** (optional): Die Position von Elementen wird verändert
- ❖ **Stilangaben** (optional) für Bildschirm und Ausdruck in 2 Dateien (`style.css` und `print.css`)
- ❖ Der Einsatz des *Bootstrap-Frameworks* ist nur in Ausnahmefällen möglich.
 - Dann müssen aber mehrere Funktionen verwendet und wesentliche Anpassungen der Standardeinstellung vorgenommen werden

Projektaufgabe

Technologien und Formate 2

Javascript

- ❖ **Formulareingaben** sind per Javascript vor der Weitergabe an den Server zu prüfen und dem Nutzer bei Fehleingaben Hinweise (per Dialog oder direkt im Formular) zu geben.
- ❖ Auswahlfelder werden durch eine **AJAX-Abfrage** vorab mit aktuellen Werten gefüllt.
- ❖ **Bibliotheken** (jQuery, etc.) sind nur **nach Absprache zulässig!**
- ❖ Elemente der Seite (Bilder, Textbereiche) erscheinen erst durch Nutzerinteraktion

Datenbank

- ❖ **SQL-Datenbank** mit **mindestens fünf Tabellen** (incl. erforderlicher Verknüpfungen über mindestens drei Ebenen)

PHP

- ❖ **Dynamischer Seitenaufbau:** Die einzelnen Abschnitte der Seite werden aus HTML- und PHP-Code zusammengefügt.
- ❖ **Häufig verwendete Seitenelemente** (Kopfzeile, Navigation, Inhalt, Fußzeile, Verbindung zur Datenbank) werden in separate HTML-bzw. PHP-Dateien ausgelagert und dynamisch mittels PHP eingebunden.

Projektaufgabe

Technologien und Formate 3

Datenabfrage

- ❖ Aus der Datenbank sollen **Daten ausgelesen** und in einer Tabelle bzw. in Formularfeldern auf der Webseite **angezeigt** werden.
- ❖ Es soll ein **Suchformular** für Produkte/Dienstleistungen/Kunden o.ä. erstellt werden, in dem mindestens **drei Suchkriterien** angegeben werden können, die mit AND oder OR verbunden werden können.

Dateneingabe

- ❖ Es sollen formularbasiert Daten **in die Datenbank eingefügt** bzw. **abgefragte Daten geändert** werden können.
- ❖ Dabei sollen bestimmte **Formularelemente ihren Wert dynamisch** zugewiesen bekommen (z.B. kaskadierende Listen-Felder oder durch Vorauswahl belegte Felder).
- ❖ Mindestens **ein Formular** soll **mehrseitig** sein und die Daten sollen **per Session weitergereicht** werden.
- ❖ Bevor die Daten in die Datenbank übernommen bzw. bei mehrseitigen Formularen auf die nächste Seite weitergegeben werden, soll eine **Prüfung auf Vollständigkeit und Plausibilität** (selbst gewählte Kriterien, mindestens drei) vorgenommen werden. Im Fehlerfall sollen die entsprechenden Formularfelder markiert werden, die schon eingegebenen Formularinhalte sollen jedoch erhalten bleiben.

Das Design der Formulare und der Ausgabe ist der Gestaltung der Website anzupassen.

Projektaufgabe

Zielplattform

Frontend

- ❖ aktueller Chrome-Browser (auf Windows 10)
- ❖ Bildschirmauflösung
 - Standard: M Medium (Desktops) > 992px
 - optional: S Small (Tablets): < 992px

Backend

- ❖ Datenbank und Webserver: XAMPP-Server 7.1 oder neuer (Apache + MySQL/MariaDB)

Projektaufgabe

Dokumentation

- ❖ **Entwicklung und Aufbau der Website** sind auf einer weiteren Unterseite, die über das Impressum verlinkt ist, zu dokumentieren. Diese Seite ist sinnvoll zu untergliedern; die grafische Gestaltung kann dagegen schlicht sein.
 - Inhalt: Beschreibung des fiktiven Kunden und seiner Ziele, sowie der späteren Websitenutzer
 - Ergebnisse der Recherche zu ähnlichen Seiten (Links, Screenshots)
 - Allgemeines Seitenlayout (Wireframe) und Verlinkung/Navigationsstruktur (Sitemap)
 - Design: Begründung zur Wahl des Layouts, der Farben, Schriften,...
 - Beschreibung der Funktionalitäten für jede Seite (beschriftete Screenshots)
 - ER-Modell und relationales Modell der Datenbank (kann als Grafik eingebunden werden)
 - Rollenmodell
 - Flussbild für Dateneingabe per mehrseitigem Formular mit Validierung
 - bewältigte Herausforderungen und Lösungen
 - Besonderheiten für die Nutzung oder Bewertung
 - Hinweise auf besondere techn. Anforderungen, bekannte Fehler, unvollständige Implementierungen, "versteckte" Zusatzfeatures...
 - Projektmanagement-Dokumentation: Auflistung von Tätigkeiten, zuständiger Person und benötigtem Aufwand

 - ❖ **Installationshinweise und Zugangsdaten** sind in einem gesonderten Text- oder PDF-Dokument zu notieren.
-

Bewertung

Kriterien

Anspruchsniveau: Umfang, Komplexität

Kenntnisse und Interesse: fundiert und interessiert

Vorgehensweise: zielgerichtet und planmäßig

Qualität der Ergebnisse: vollständig, richtig und nachvollziehbar

❖ "sauberer" Code (HTML, CSS, JS, PHP)

- standardkonforme und konsistente Struktur und Gliederung
- sinnvolle Aufteilung der Dateien und Ordnerstruktur (nach Vorgabe)
- durchgehender Codestyle (nach Vorgabe) und sinnvolle Kommentierung für leichte Nachvollziehbarkeit
- sprechende Namen für Dateien, Stile, Variablen, Funktionen etc.
- keine überflüssigen Dateien oder Code-Bestandteile
 - also gut aufräumen und (auf anderem Rechner) testen: Browserkonsole beobachten, Validator,...!

❖ grafische Gestaltung und Bedienbarkeit

- klare Gestaltung, passende Farbigkeit
- gute Lesbarkeit, verständliche Navigation, gute Bedienbarkeit
- Nutzbarkeit auf 2 versch. Bildschirmauflösungen/Fenstergrößen

Präsentation + Dokumentation

Projektaufgabe

Gruppenarbeit

- ❖ Die Bearbeitung erfolgt in **2er-Teams!**
- ❖ Alle Mitglieder müssen am Ende ein **gutes Verständnis von allen Aspekten der Website** haben und Fragen dazu beantworten können.
- ❖ Die **Arbeitsteilung** und der jeweilige **Aufwand** ist bei der Präsentation kurz und bei der Endabgabe ausführlich zu dokumentieren.

Zusammenarbeit

- ❖ Legen Sie eine Austauschplattform fest, so dass alle immer auf dem gleichen Stand sind. So haben Sie auch gleich ein Backup.
- ❖ Sammeln Sie Materialien in einer sinnvollen Verzeichnisstruktur:
„Ordnung ist das halbe Projekt“
- ❖ Erstellen Sie von Beginn an ein Dokument mit allen Infos und pflegen Sie dieses regelmäßig.
- ❖ Online-Tools wie *Trello* helfen sehr, den Überblick zu wahren.
- ❖ Aber: Treffen Sie sich unbedingt persönlich und arbeiten Sie nebeneinander.

Projektaufgabe

Herangehensweise 1

Aufgabenstellung aufmerksam durchlesen und ggf. Rückfragen stellen

Themenfindung

- ❖ Branche, Anbieter und Inhalte grob festlegen

Recherche und Ideensammlung

- ❖ Was gibt es für ähnliche Webseiten? > Screenshots erstellen
- ❖ Welche Funktionen sind sinnvoll? Was sieht für die Kunden passend aus?
- ❖ Beschreiben Sie erstmal alle(!) Ideen für Inhalte, Layouts, Navigation usw.
 - Arbeiten Sie dabei auch (vorläufig) analog mit Papier und Stift.

Entwurf

- ❖ Bewerten Sie die gesammelten Ideen. Ggf. lassen diese sich auch kombinieren.
 - Gleichen Sie die Aspekte mit der Aufgabenstellung ab.
 - ❖ Skizzieren Sie mögliche Layouts und Navigationsstrukturen.
 - ❖ Notieren Sie Herausforderungen und offene Fragen.
 - ❖ Besprechen Sie den Entwurf auf Grundlage einer textuellen und bildlichen Beschreibung mit den Dozenten. Notieren Sie die Hinweise und Ergebnisse der Konsultation.
-

Projektaufgabe

Herangehensweise 2

Implementierung

- ❖ Teilen Sie die Aufgaben untereinander auf.
- ❖ Weisen Sie allen Aufgaben eine Priorität zu ... und halten Sie sich daran.
- ❖ Fragen Sie bei Problemen rechtzeitig(!) die Dozenten oder Kommilitonen.

Dokumentation

- ❖ Erstellen Sie von Beginn an ein oder mehrere Dokumente mit allen aktuellen Infos und pflegen Sie diese regelmäßig. So haben Sie gleich eine „wachsende“ Grundlage für die abschließende Dokumentation.
- ❖ Dieses Dokument und weitere Materialien sind essentielle Grundlage für Besprechungen mit den Dozenten. Dabei geht es erstmal noch nicht um die Form der Dokumentation!
- ❖ Planen Sie aber am Schluss ausreichend Zeit für die abschließende Dokumentation ein. Denn neben dem funktionierenden Projekt und dessen Source-Code ist die Form wichtig für den Eindruck, ob strukturiert vorgegangen wurde.
- ❖ Wenn Dinge nicht funktionieren wie gewollt oder gefordert, ist es oft gut zumindest zu beschreiben, was man alles probiert hat und was dabei nicht funktioniert hat, als gar nichts zu erwähnen.

Kontakt

Prof. Kruse

- rolf.kruse@fh-erfurt.de
- Skype: enclued

Kristof Friess

- Kristof.friess@fh-erfurt.de

Fragen, die alle betreffen könnten, bitte über das Nachrichtenforum des Moodle-Kurses stellen



The screenshot shows a Moodle course page for 'Kurs Programmierung Webanw'. The page has a blue header with the text 'UNIVERSITY OF APPLIED SCIENCES'. Below the header, there is a section titled 'Kurs Programmierung Webanw' with a 'Nachrichtenforum' (Message Forum) icon. To the left of the forum, there is a sidebar with a '16 er/innen ungen' (16 users online) indicator. Below the forum, there is a section titled 'Montag, 5. Oktober' with a list of events: '10:00 Uhr Auftaktveranstaltung' (10:00 AM Kick-off event) and '12:00 - 14:00 Uhr HTML' (12:00 - 14:00 PM HTML). The list of events includes: 'Organisatorisches', 'Einführung in HTML und PHP', and 'Einführung in die Übungsumr'.