



alma-

Templatesystem für Institutshomepages

Um universitäre Websites effektiv verwalten zu können, baut alma auf dem Model-View-Controller-Konzept auf. Dies ermöglicht ein leichtes Einbinden vorhandener Verwaltungsmodule, eine vollständige Trennung von Code und Design und eine gute Standardkonformität z.B. in Bezug auf Barrierefreiheit. Die Pflege der Inhalte setzt keinerlei Programmierkenntnisse voraus, ohne daß hierfür ein komplexes Redaktionssystem notwendig ist. Realisiert wird alma in der Skriptsprache Perl, wobei bewährte CPAN-Module wie **Class::DBI**, **Template Toolkit** und **Catalyst** verwendet werden.

Universitäre Institute müssen sich heute selbstverständlich im Internet präsentieren. Dabei sind Seriosität und Aktualität der Inhalte genauso gefordert wie ein standardkonformer, barrierefreier Zugriff. So klar aber das Anforderungsprofil ist, so wenig einheitlich sind die Umsetzungs- und Betreuungsstrategien.

Das von der Medienkommission des Akademischen Senats geförderte Projekt alma verfolgt zwei Ziele: erstens die Programmierung eines standardkonformen Websitetemplates nach dem MVC-Modell am Beispiel des Kulturwissenschaftlichen Seminars und zweitens die exemplarische Einbindung bereits vorhandener Funktionsmodule wie **pbib** für die Literaturverwaltung oder **hora** für die Koordination von Lehrveranstaltungen.



Das Templatesystem
Ein Templatesystem trennt Websites funktional nach Inhalt, Design und Logik. Nur so lassen sich die Inhalte leicht – mithilfe intuitiver WYSIWYG-Editoren – aktualisieren oder neue Programmfunktionalitäten einbauen. Das MVC-Modell systematisiert genau diese Anforderungen. MVC steht für Model, View und Controller, wobei der Controller für die Steuerung der Applikation, das Model für Datenhaltung und die View für die grafische Darstellung zuständig ist.

Ein typischer Funktionsablauf ist der Login. Ein User sendet seine Daten über eine View, etwa ein HTML-Formular mit Username und Paßwort, an den Controller. Dieser prüft die Daten und schickt sie weiter an das Model. Dieses schaut in einer MySQL-Datenbank nach, ob der User existiert und das Paßwort korrekt ist. Das Ergebnis wird dem Controller mitgeteilt. Im positiven Falle leitet der Controller den User eine Seite in der Application weiter, im negativen Falle wird er zurück zur Loginseite geschickt.

Der Vorteil dieser abstrakten Modellierung liegt in der Flexibilität und Wiederverwendbarkeit des Codes. Es reicht beispielsweise aus, die View zu verändern, um die Applikation von HTML auf XML oder WAP umzustellen. Natürlich verkraftet das MVC-Konzept auch ein Redesign oder die Umstellung der Datenbank von MySQL auf LDAP ohne größere Probleme.

Als Middleware verwendet alma die Skriptsprache Perl. Das von Sebastian Riedel initiierte Perl-Modul Catalyst stellt eine ideale Web Application Framework für das MVC-Modell dar. Das Konzept baut auf folgenden bewährten Perl-Modulen mit entsprechender funktionaler Differenzierung auf:

- das Datenmodell (Model): **Class::DBI** für den Zugriff und die Änderung von Inhalten (Daten),
- Darstellungsschicht (View): **Template-Toolkit** für die Darstellung der Inhalte,
- die Steuerungsschicht (Controller): **Catalyst** für die Kontrolle aller Anforderungen, Parameterüberprüfung usw.



Die Barrierefreiheit
Auf der View-Seite wird standardkonformes XHTML 1.0 generiert, womit bereits ein hinreichendes Maß an Barrierefreiheit realisiert wird, insofern sämtliche Funktionselemente zwingend eindeutig benannt werden müssen. Das Design wird – soweit es überhaupt möglich ist – über Style Sheets realisiert und ist möglichst funktional gehalten. Es gibt weder Frames noch Multimediaelemente, die alternativlos verwendet werden müssen. Ein gutes ‚Bild‘ einer Seite, die nicht visuell wahrgenommen wird, erhält man durch Ausschalten der Style Sheets bzw. durch Analyse mittels terminalbasiertem Browser wie lynx.

Die Funktionsmodule
Das Kulturwissenschaftliche Seminar verbindet medienpraktische Lehr- und Lernanliegen immer auch mit der Entwicklung von Funktionsmodulen, die allgemein verwendbar sind. Am Beispiel von alma soll gezeigt werden, wie sich derartige Module in eine Institutswebsite einbinden lassen. Verwaltet werden dabei u. a.:

- hora: Institutionen, Kollegium, Lehrveranstaltungen
- pbib: Bibliographien, Volltexte
- hyperion: Projekte, Hausarbeiten
- nodus: IT-Geräte, Netzwerkdaten.



HU Berlin | Kulturwissenschaftliches Seminar
Dr. Christian Kassung | 2093-8240 | CKassung@culture.hu-berlin.de
<http://admin.culture.hu-berlin.de/>

Multimedia-Förderprogramm 2004

