

7. Fazit

Abschließend betrachtet stellt das AJAX- Konzept eine sehr interessante Möglichkeit zur Erstellung interaktiver Webseiten dar. Es kann in allen gängigen Browsern eingesetzt werden und stellt keine neuen Herausforderungen an die Hard- und Software von Server und Client. Aufbauend auf bereits bekannten Technologien lässt sich auch der Umgang mit AJAX relativ leicht erlernen, einfache Beispiele lassen sich innerhalb von Minuten erstellen. Gleichzeitig bietet es aber genügend Werkzeuge und Freiraum um auch komplexe Projekte umzusetzen.

Für den Benutzer stellt AJAX ebenfalls eine Bereicherung dar. Das Arbeiten mit AJAX Anwendungen erscheint flüssiger, Wartezeiten entfallen. Mit Internetapplikationen kann man nun genauso wie mit herkömmlichen, lokal installierten Programmen arbeiten.

Allerdings birgt das Konzept derzeit noch einige Schwächen. Die nicht einheitliche Erzeugung des XMLHttpRequest- Objekts verkompliziert die Erstellung eines universal lauffähigen AJAX- Skripts, die Verwendung von JavaScript oder gar ActiveX- Komponenten verunsichert sicherheitskritische Programmierer. Zum Glück untersagen die Browser dem XMLHttpRequest- Objekt, mit einer anderen als der eigenen Domäne zu kommunizieren, ansonsten müsste man befürchten, dass AJAX vor allem dazu verwendet würde, unerlaubt Benutzerdaten zu sammeln und zu versenden. Das verwehrt dem Konzept aber auch den direkten Zugriff auf externe Webservices für die Informationseinholung, die heute eine wichtige Rolle im Internet spielen. Es ist schwierig AJAX barrierefrei zu implementieren und Barrierefreiheit ist eine der Grundvoraussetzungen bei öffentlichen Ausschreibungen. Eine Suchmaschine wird die dynamisch bereitgestellten Informationen in einer AJAX- Applikation niemals aufspüren können.

Man sollte also sparsam mit dem Einsatz von AJAX umgehen und es nur dort zum Einsatz kommen lassen, wo es wirklich Vorteile bringt. Der derzeitige Hype ist wohl nur ein vorübergehendes Phänomen. Vieles lässt sich ohne AJAX einfacher umsetzen. Aber die Entwicklung ist noch nicht zu Ende. Bereits seit einiger Zeit arbeitet das W3C an neuen AJAX- Standards, die die bekannten Schwächen ausbügeln und die Handhabung insgesamt vereinfachen soll.

AJAX ist mit Sicherheit keine Killerapplikation, die das Internet revolutioniert, es ist nur ein alternativer Programmieransatz bei der Kommunikation zweier Computer. Aber zusammen mit anderen Technologien und Konzepten des Web

2.0 eröffnet sich für Benutzer und Programmierer eine neue Welt jenseits von HTML mit Hyperlinks. Dabei ist AJAX gar kein neues Konzept. Nur das Umfeld hat sich so geändert, dass der Einsatz sinnvoll erscheint.

Das Praxisbeispiel kann man durchaus als eine gelungene Umsetzung des AJAX-Konzepts bezeichnen. Die Integrierung in bestehende Formulare lässt sich mit wenigen Zeilen Code umsetzen, sobald das Wörterbuch in der Datenbank und die Skripte auf dem Server installiert sind. Wenn man einmal davon absieht, dass die vorgefertigten Funktionen von PHP und MySQL linguistisch nicht perfekt für die deutsche Sprache konzipiert sind, läuft die Rechtschreibprüfung zuverlässig und fehlerfrei ab. Vom Ablauf und Layout her an bestehende Korrekturprogramme angelehnt, lässt sich die Benutzeroberfläche intuitiv bedienen. Da die Vorteile des AJAX-Konzepts konsequent genutzt werden, laufen die Vorgänge im Programm fließend ab, es entstehen nur kurze Wartezeiten. Man hätte das Praxisbeispiel auch ohne AJAX programmieren können, allerdings mit deutlich mehr Aufwand und nicht ohne große Änderungen am eigentlichen Fragebogen.

Der einzige wirkliche Wermutstropfen des Praxisbeispiels liegt darin, dass zunächst Pop-Up-Fenster für die Webseite erlaubt werden müssen, bevor sich das Fenster zur Korrektur des Textes öffnet, ein Schritt, den in Zeiten lästiger Werbe-Pop-Ups nicht jeder Benutzer gerne macht. Auch ist die Aktivierung von JavaScript Pflicht, sollte es im Browser des Benutzers deaktiviert sein, kann er aber trotzdem noch einwandfrei mit dem Fragebogen arbeiten.

Im konkreten Kontext des aievas Projektmanagementsystems lässt sich das Praxisbeispiel nach ein paar Anpassungen stilistischer Art nahtlos einfügen. Eine mehrsprachige Umsetzung ist mit wenig Aufwand zu erreichen. Typische Anfragen in Form von Kommentartexten im Rahmen der Gegebenheiten des aievas-Systems sollten die eingesetzten Server meistern können, wie ausgiebige Tests vor allem am Flaschenhals des Konzepts, der MySQL-Datenbank gezeigt haben. Wenn man die Performance mit anderen Online-Rechtschreibprüfungen vergleicht, zum Beispiel der in Writely (die in Kapitel 2 vorgestellte AJAX-Textverarbeitung), stellt man fest, dass das Praxisbeispiel ähnliche Geschwindigkeiten beim Erkennen falscher Wörter aufweist.

Bleibt nur noch die Frage, ob die aievas-Mysteryshopper die neue Funktion auch annehmen und benutzen werden. Da die Qualitätssteigerung der abgelieferten Texte auch in Interesse der Benutzer liegt und eine Online-Korrektur nicht viel Zeit erfordert bin ich überzeugt davon, dass die Benutzer

regen Gebrauch von der integrierten Rechtschreibprüfung machen werden und so das Ziel der Aufwandsverringerung bei der Aufbereitung der Rohdaten erreicht wird.