

Hochschule Heilbronn
Technik – Wirtschaft – Informatik

Studiengang Electronic Business

Diplomarbeit

"Java-Frameworks für Rich Internet Applications"

vorgelegt bei
Professor Dr. Michael Gröschel

von
Tushig Tumenjargal
Matr.-Nr. 160924

im
Sommersemester 2008

5.5	JavaFX	- 63 -
5.5.1	Hintergrund	- 63 -
5.5.2	Architektur	- 63 -
5.5.3	JavaFX Script	- 64 -
5.5.4	Systemvoraussetzungen.....	- 64 -
5.5.4.1	Entwicklung.....	- 64 -
5.5.4.2	Client	- 64 -
5.5.5	Installation und Konfiguration	- 64 -
5.5.6	Kosten.....	- 64 -
5.5.7	Skill-Anforderungen.....	- 65 -
5.5.8	Stabilität und Reife.....	- 65 -
5.5.9	Features und Funktionsumfang	- 65 -
5.5.10	Dokumentation und Support	- 65 -
5.5.11	Zusammenfassung und Ausblick.....	- 66 -
5.6	Google Gears	- 66 -
5.6.1	Hintergrund	- 66 -
5.6.2	Funktionsweise.....	- 67 -
5.6.3	Ausblick	- 68 -
5.7	Bewertung	- 69 -
5.7.1	Szenario 1	- 69 -
5.7.2	Szenario 2	- 70 -
5.8	Zusammenfassung	- 70 -
6	Fazit und Ausblick	- 72 -
	Danksagung	X
	Literaturverzeichnis	XI
	Internetquellen	XIII
	Ehrenwörtliche Erklärung	XV

Inhaltsverzeichnis

Inhaltsverzeichnis	II
Abbildungsverzeichnis.....	VI
Abkürzungsverzeichnis	VII
Management Summary	VIII
1 Einleitung.....	- 1 -
1.1 Zielsetzung	- 1 -
1.2 Vorgehensweise	- 2 -
2 Grundlagen.....	- 3 -
2.1 Client/Server-Architekturen.....	- 3 -
2.1.1 Logische Schichtung	- 3 -
2.1.2 Physische Schichtung.....	- 4 -
2.1.3 2-Tier-Architektur	- 4 -
2.1.4 3-Tier-Architektur	- 5 -
2.1.5 n-Tier-Architektur	- 5 -
2.1.6 Thin-Client-Modell	- 6 -
2.1.7 Rich-Client-Modell	- 7 -
2.2 Clientseitige Web-Technologien	- 8 -
2.2.1 Java Applets	- 8 -
2.2.2 Java Web Start.....	- 8 -
2.2.3 JavaScript/Ajax	- 9 -
2.2.4 ActiveX	- 9 -
2.2.5 Flash/ActionScript.....	- 10 -
2.3 Frameworks in der Web-Anwendungsentwicklung	- 11 -
2.3.1 Was ist ein Framework?	- 11 -
2.3.2 Warum Frameworks?	- 11 -
2.3.3 Model View Controller	- 12 -
2.4 Zusammenfassung	- 14 -
3 Rich Internet Applications: Definition, Einsatzbereiche, Technologien	- 15 -
3.1 Was sind Rich Internet Applications?	- 15 -
3.2 RIA-Architektur	- 17 -
3.3 Einsatzbereiche	- 18 -
3.4 Anforderungen an RIAs	- 19 -
3.5 RIA-Technologien.....	- 20 -
3.6 Zusammenfassung	- 22 -
4 Beurteilungsrahmen für RIA-Frameworks.....	- 23 -
4.1 Interessensgruppen in der RIA-Entwicklung	- 24 -
4.1.1 Benutzer	- 24 -
4.1.2 Entwicklung	- 24 -
4.1.3 Betrieb	- 24 -
4.2 Kriterienkatalog.....	- 25 -
4.2.1 Systemvoraussetzungen.....	- 25 -

4.2.2	Kosten.....	- 25 -
4.2.3	Skill-Anforderungen.....	- 26 -
4.2.4	Stabilität und Reife.....	- 26 -
4.2.5	Features und Funktionsumfang.....	- 26 -
4.2.6	Dokumentation und Support.....	- 26 -
4.2.7	Wartbarkeit.....	- 26 -
4.2.8	Performance.....	- 26 -
4.2.9	Sicherheit.....	- 27 -
4.2.10	Zusammenfassung.....	- 27 -
4.2.11	Ausblick.....	- 27 -
4.3	Beispielszenarien.....	- 27 -
4.3.1	Beispielszenario 1.....	- 27 -
4.3.2	Beispielszenario 2.....	- 28 -
4.4	Zusammenfassung.....	- 28 -
5	Evaluierung von RIA-Frameworks.....	- 29 -
5.1	Google Web Toolkit.....	- 29 -
5.1.1	Hintergrund.....	- 29 -
5.1.2	Architektur.....	- 30 -
5.1.2.1	Java-to-JavaScript Compiler.....	- 30 -
5.1.2.2	Hosted Web Browser.....	- 30 -
5.1.2.3	Die JRE-Emulationsbibliothek.....	- 31 -
5.1.2.4	GWT-Web-UI-Klassenbibliothek.....	- 32 -
5.1.3	GWT-Module.....	- 32 -
5.1.4	Systemvoraussetzungen.....	- 32 -
5.1.4.1	Entwicklung.....	- 32 -
5.1.4.2	Clientseite.....	- 33 -
5.1.5	Installation und Konfiguration.....	- 33 -
5.1.6	Kosten.....	- 34 -
5.1.7	Skill-Anforderungen.....	- 34 -
5.1.8	Stabilität und Reife.....	- 34 -
5.1.9	Features und Funktionsumfang.....	- 34 -
5.1.10	Dokumentation und Support.....	- 36 -
5.1.11	Wartbarkeit.....	- 37 -
5.1.12	Performance.....	- 37 -
5.1.13	Sicherheit.....	- 38 -
5.1.14	Zusammenfassung.....	- 38 -
5.1.15	Ausblick.....	- 39 -
5.2	JBoss RichFaces.....	- 39 -
5.2.1	Hintergrund.....	- 39 -
5.2.2	Architektur.....	- 40 -
5.2.2.1	Ajax Filter.....	- 40 -
5.2.2.2	Ajax Action Components.....	- 42 -
5.2.2.3	Ajax Containers.....	- 42 -
5.2.2.4	JavaScript Engine.....	- 42 -
5.2.2.5	Skinnability.....	- 42 -
5.2.3	Systemvoraussetzungen.....	- 42 -
5.2.3.1	Entwicklung.....	- 42 -
5.2.3.2	Clientseite.....	- 43 -

5.2.4	Installation und Konfiguration	- 43 -
5.2.5	Kosten.....	- 44 -
5.2.6	Skill-Anforderungen.....	- 44 -
5.2.7	Stabilität und Reife.....	- 44 -
5.2.8	Features und Funktionsumfang	- 45 -
5.2.9	Dokumentation und Support	- 46 -
5.2.10	Wartbarkeit.....	- 47 -
5.2.11	Performance	- 47 -
5.2.12	Sicherheit.....	- 48 -
5.2.13	Zusammenfassung	- 48 -
5.2.14	Ausblick	- 49 -
5.3	OpenLaszlo	- 49 -
5.3.1	Hintergrund	- 49 -
5.3.2	Architektur	- 49 -
5.3.2.1	Client	- 50 -
5.3.2.2	Server.....	- 50 -
5.3.2.3	Deployment	- 51 -
5.3.2.4	LZX	- 52 -
5.3.3	Systemvoraussetzungen.....	- 53 -
5.3.3.1	Entwicklung.....	- 53 -
5.3.3.2	Clientseite	- 53 -
5.3.4	Installation und Konfiguration	- 53 -
5.3.5	Kosten.....	- 53 -
5.3.6	Skill-Anforderungen.....	- 54 -
5.3.7	Stabilität und Reife.....	- 54 -
5.3.8	Features und Funktionsumfang	- 54 -
5.3.9	Dokumentation und Support	- 56 -
5.3.10	Wartbarkeit.....	- 56 -
5.3.11	Performance	- 56 -
5.3.12	Sicherheit.....	- 57 -
5.3.13	Zusammenfassung	- 57 -
5.3.14	Ausblick	- 57 -
5.4	Java Web Start mit Swing.....	- 58 -
5.4.1	Hintergrund	- 58 -
5.4.2	Funktionsweise.....	- 58 -
5.4.3	Swing.....	- 59 -
5.4.4	Systemvoraussetzungen.....	- 59 -
5.4.4.1	Entwicklung.....	- 59 -
5.4.4.2	Clientseite	- 59 -
5.4.5	Installation und Konfiguration	- 59 -
5.4.6	Kosten.....	- 59 -
5.4.7	Skill-Anforderungen.....	- 60 -
5.4.8	Stabilität und Reife.....	- 60 -
5.4.9	Features und Funktionsumfang	- 60 -
5.4.10	Dokumentation und Support	- 61 -
5.4.11	Wartbarkeit.....	- 61 -
5.4.12	Performance	- 61 -
5.4.13	Sicherheit.....	- 62 -
5.4.14	Zusammenfassung	- 62 -
5.4.15	Ausblick	- 63 -

Management Summary

Der ursprüngliche Einsatzbereich von HTML – die Präsentation vernetzter Dokumente – wurde in den letzten Jahren stark erweitert bis hin zur interaktiven Präsentation von multimedialen Inhalten oder den grafischen Benutzerschnittstellen von Web-Anwendungen. Der Browser hat sich zu einem universellen Client entwickelt: Immer mehr Geschäftsanwendungen werden in browserbasierte Anwendungen oder Portal-Lösungen migriert.

Komplexe Anwendungen benötigen jedoch viel mächtigere Bedienoberflächen als es allein mit HTML realisiert werden kann, und so gab es bis vor wenigen Jahren Bereiche, in denen Web-Anwendungen nicht mit klassischer Software konkurrieren konnten. Beispielsweise konnten Internet-Applikationen dem Online-Nutzer weder ausreichend hohe Interaktivität noch reichhaltige Benutzerschnittstelle anbieten.

Browserbasierte Rich Internet Applications (RIA) sind ein möglicher Ausweg aus diesen sich widersprechenden Anforderungen an Komplexität und Bedienungskomfort. RIAs bereichern Inter- und Intranet-Auftritte durch visuelle Effekte und verbessern zusätzlich die Interaktionsmöglichkeiten. RIAs versuchen, die Kluft zwischen den klassischen Web- und Desktop-Applikationen zu schließen, indem sie die Vorteile von beiden vereinen. Ziel dabei ist es, Web-Applikationen mit Rich-Client-Eigenschaften bereitzustellen.

Für die Realisierung von RIA-Lösungen bieten sich mehrere Technologien an, die jeweils in unterschiedlichen Bereichen ihre Stärken besitzen. Für geschäftskritische Anwendungen empfiehlt es sich, auf etablierte Standardtechnologien zu setzen. Zu solchen Technologien können die Java-Plattform von Sun oder die .NET-Plattform von Microsoft gezählt werden. Mit Adobe Flash existiert eine Plattform, die häufig in multimedial geprägten Web-Auftritten eingesetzt wird. Darüber hinaus gibt es verschiedene RIA-Lösungen, die auf proprietäre Skriptsprachen bzw. diverse XML-Derivate setzen, um die Erstellung von reichhaltigen Benutzerschnittstellen zu ermöglichen.

Eine große Bedeutung für die RIA-Entwicklung hat Ajax als Schlüsseltechnik für die Realisierung der partiellen Seitenerzeugung (Partial Page Rendering) bzw. asynchronen Datenverarbeitung in browserbasierten Applikationen. AJAX steht für Asynchronous JavaScript and XML und stellt eine Kombination aus den Webstandards JavaScript, XML, HTML und CSS dar.¹ Als Mittel zur Verbesserung der Benutzerinteraktion ist Ajax aus den heutigen Web-Applikationen nicht mehr wegzudenken.

Die Entscheidung für eine bestimmte Technologie hängt von den jeweiligen Anforderungen und nicht zuletzt von der bestehenden Infrastruktur ab. In der Fachliteratur wird der Einsatz der Java-Plattform für geschäftliche Applikationen empfohlen. Dies wird damit begründet, dass die Java-Plattform gleich mehrere Anforderungen an die Industriequalität erfüllt und auf den Standards basiert.

¹ Vgl. WWW, Ajax: A New Approach to Web Applications. 12.06.2008.

Wenn Web-Applikationen mit neuen Technologien realisiert werden sollen, sollten diese zunächst mit kritischem Auge betrachtet werden. Zur Bewertung, ob und wie ein neuer Ansatz im Kontext kritischer Softwareprodukte funktioniert, sollten die Potenziale und Risiken rational bewertet werden. Ein guter Einstieg in eine solche Betrachtung ist meist der Blick auf die vorhandene Infrastruktur. Dazu zählen eingesetzte Plattformen, Server und Umgebungen genauso wie der Blick auf die verfügbaren Clients.

RIA-Lösungen bedeuten allerdings im Vergleich zu klassischen Web-Anwendungen nicht nur eine funktionsreichere Benutzerschnittstelle, sondern auch einen höheren Implementierungsaufwand. Deshalb sind Werkzeuge wichtig, die den Entwicklungsprozess unterstützen, indem sie bspw. bereits vorgefertigte Komponenten zur Verfügung stellen oder die Komplexität verbergen. Es existiert aktuell eine große Vielfalt an unterstützenden Frameworks für die Entwicklung von RIA-Lösungen. Daran kann die Bedeutung der RIA-Technologie generell und das Interesse seitens der Web-Entwicklungsgemeinde an neuartigen Internet-Applikationen abgeleitet werden.

Doch diese Vielfalt an verfügbaren Frameworks macht deren Auswahl für eigene Projekte schwierig. Es fällt den Entwicklern schwer, den Überblick im entstandenen „Dschungel“ der Frameworks und Komponenten-Bibliotheken zu behalten. Es ist deshalb wichtig, über einen systematischen Ansatz für einen Entscheidungsprozess in RIA-Projekten zu verfügen. Ein solcher Ansatz sollte eine Empfehlung für die grundlegende Vorgehensweise und Methodik der Evaluierung und Auswahl eines geeigneten Frameworks liefern.

Die so ausgearbeitete Vorgehensweise sollte laufend in realen Projekten auf ihre Praxistauglichkeit getestet und notfalls ständig optimiert werden. Analysiert und verbessert werden soll vor allem die Art und Weise, wie man an eine solche Problematik herangeht. Zusätzlich sollten eigene Analysemethoden kontinuierlich an aktuelle Technologien und Anforderungen angepasst und als unterstützendes Werkzeug weiterentwickelt werden.