



HTWK Leipzig

Fakultät für Informatik, Mathematik & Naturwissenschaften

Bachelor-Thesis

Generierung und Design einer Client-Bibliothek für einen RESTful Web Service am Beispiel der Spreadshirt-API

Author:

Andreas Linz

10INB-T

admin@klingt.net

Leipzig, 13. Juni 2013

Gutachter:

Dr. rer. nat. Johannes Waldmann

HTWK Leipzig

Fakultät für Informatik, Mathematik & Naturwissenschaften

waldmann@imn.htwk-leipzig.de

HTWK Leipzig, F-IMN, Postfach 301166, 04251 Leipzig

Andreas Linz
Nibelungenring 52
04279 Leipzig
admin@klingt.net
www.klingt.net

*Generierung und Design einer Client-Bibliothek für einen
RESTful Web Service am Beispiel der Spreadshirt-API*
Bachelor Thesis, HTWK-Leipzig, 13. Juni 2013

made with X_YTEX, L^AT_EX and B^IB_TE_X.

Selbstständigkeitserklärung

Ich erkläre hiermit, dass ich diese Bachelor-Thesis selbstständig ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel verfasst habe. Alle den benutzten Quellen wörtlich oder sinngemäß entnommenen Stellen sind als solche einzeln kenntlich gemacht. Diese Arbeit ist bislang keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht worden.
Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben wird.

.....
Andreas Linz
Leipzig, 13. Juni 2013

Danksagungen

Mein Dank gilt Dr. rer. nat. Johannes Waldmann für die Betreuung der Arbeit, sowie Spreadshirt und insbesondere Jens Hadlich für die Unterstützung.

Ohne das Korrekturlesen von Elisa Jentsch hätte ich so manchem Fehler nicht entdeckt, dafür auch noch einmal vielen Dank.

Abstract

Schlüsselwörter

Codegenerierung, RESTful Web Service, Modellierung, Client-Bibliothek, Spreadshirt-API, Polyglot

Diese Seite wurde mit Absicht leer gelassen.

Inhaltsverzeichnis

1	Einführung	1
1.1	Spreadshirt	1
1.2	Warum Code generieren?	2
1.2.1	Hello	2
2	Grundlagen	4
2.1	RESTful Web Service	4
2.2	WADL	4
2.3	XSD	4
3	Implementierung	5
4	Zusammenfassung	6
4.1	Fazit	6
4.2	Ausblick	6
	Glossary	A
	Abbildungsverzeichnis	A
	Tabellenverzeichnis	B
	Literatur	C

Diese Seite wurde mit Absicht leer gelassen.

1 Einführung

Dies ist ein Zitat

Andreas Linz

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

1.1 Spreadshirt

Das hier ist der zweite Absatz. Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft.

Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

1.2 Warum Code generieren?

Und nun folgt – ob man es glaubt oder nicht – der dritte Absatz. Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

1.2.1 Hello

Nach diesem vierten Absatz beginnen wir eine neue Zählung. Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt

sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

1.2.1.1 World

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

2 Grundlagen

2.1 RESTful Web Service

2.2 WADL

2.3 XSD

3 Implementierung

4 Zusammenfassung

4.1 Fazit

4.2 Ausblick

Glossary

foo bar. A

Abbildungsverzeichnis

Tabellenverzeichnis

Literatur

- Czarnecki, K. und U. Eisenecker (2000). *Generative programming: methods, tools, and applications*. Addison Wesley. ISBN: 9780201309775. URL: <http://books.google.de/books?id=cZXYQ6Pau4C>.
- Fowler, M. (2010). *Domain-Specific Languages*. Addison-Wesley Signature Series (Fowler). Pearson Education. ISBN: 9780131392809. URL: http://books.google.de/books?id=ri1muolw_YwC.
- Herrington, J. (2003). *Code Generation in Action*. In Action Series. Manning. ISBN: 9781930110977. URL: <http://books.google.de/books?id=VHVC8WnSgbyC>.
- Kelly, S. und J.P. Tolvanen (2008). *Domain-Specific Modeling: Enabling Full Code Generation*. Wiley. ISBN: 9780470249253. URL: http://books.google.de/books?id=GFFtRFkuU_AC.
- Klar, M. und S. Klar (2006). *Einfach generieren: Generative Programmierung verständlich und praxisnah*. Hanser Fachbuchverlag. ISBN: 9783446404489. URL: <http://books.google.de/books?id=6LS70wAACAAJ>.