

Dokumentation - Blog-Service

Webbasierte Anwendung mit
asynchroner Client-Server Kommunikation
aus der
Benutzer- und Implementierungssicht

LVA: KV Service Engineering

Übung 1/2

Gruppe3

Strahinja Lazic	k1255095
Andreas Lengauer	k1455988
Lukas Mindlberger	k1255809
Robert Ördög	k1355928
Christian Reiter	k1355821

Inhaltsverzeichnis

1. Einleitung	3
1.1 Aufgabenstellung	3
1.2 Begriffserklärungen/ Implementierungsparameter	3
1.2.1 Blogservice:	3
1.2.2 Javascript:	3
1.2.3 JSON:	4
1.2.4 NodeJS:.....	4
1.2.5 MySQL:.....	4
1.2.6 HTML:	4
1.2.7 AngularJS:	5
1.2.8 HTTP:	5
2. Implementierungssicht	5
3. Benutzersicht	5
3.1 Erstellen eines Posts.....	6
3.2 Anzeigen von Posts.....	6
3.3 Erstellen von Gruppen	6

1. Einleitung

1.1 Aufgabenstellung

Ziel dieser Übung war es eine webbasierte Anwendung zu erstellen mit asynchroner Client-Server-Kommunikation. Unserer Gruppe wurde die Erstellung eines Blogservices zugewiesen, der für die Kommunikation in offenen oder geschlossenen Freundesgruppen, inkl. Kommentarfunktion, dienen soll. Client-seitig wurde *Javascript* mit der Bibliothek *Angular* implementiert. Server-seitig wurde *Javascript* auf Basis von *NodeJS* implementiert. Die Datenhaltung erfolgt mittels *mySQL*.

1.2 Begriffserklärungen/ Implementierungsparameter

1.2.1 Blogservice:

Häufig ist ein Blog eine chronologisch abwärts sortierte Liste von Einträgen, die in bestimmten Abständen umbrochen wird. Der Blogger ist Hauptverfasser des Inhalts, und häufig sind die Beiträge aus der Ich-Perspektive geschrieben. Das Blog bildet ein Medium zur Darstellung von Aspekten des eigenen Lebens und von Meinungen zu spezifischen Themen, je nach Professionalität bis in die Nähe einer Internet-Zeitung mit besonderem Gewicht auf Kommentaren. Oft sind auch Kommentare oder Diskussionen der Leser über einen Artikel möglich. Damit kann das Medium sowohl dem Ablegen von Notizen in einem Zettelkasten, dem Zugänglichmachen von Informationen, Gedanken und Erfahrungen, etwas untergeordnet auch der Kommunikation dienen, ähnlich einem Internetforum.

1.2.2 Javascript:

Javascript ist eine Skriptsprache welche in Web Browsern eingesetzt wird, um dynamische HTML Seiten zu ermöglichen. Damit ist es möglich dynamisch Inhalte im Client Browser zu generieren bzw. zu adaptieren. Dies ermöglicht es Interaktionen mit einer Browserumgebung und den Dokumenten welche in diesem dargestellt wird zur Verfügung zu stellen. JavaScript entwickelte sich aus dem ECMA Script, welcher ein standardisierter Sprachkern ist. JavaScript ermöglicht sowohl objektorientierte, prozedural als auch funktional zu programmieren. Mit JavaScript ist es möglich HTML Elemente direkt in ein statisches HTML zu erstellen sowie existierende HTML Elemente zu manipulieren. Zusätzlich ist es möglich Events mithilfe von JavaScript zu definieren um diese HTML Elemente zu beeinflussen und Interaktionen mit dem Benutzer zu ermöglichen. JavaScript bietet als Programmiersprache Konstrukte wie Schleifen, Anweisungen sowie Fehlerbehandlungen an. Zusätzlich werden einfache sowie komplexe Datentypen unterstützt wie Objekte und Arrays. Zu beachten ist das JavaScript jedoch hoch dynamisch ist, wobei eine JavaScript Variable mehrere Datentypen beinhalten kann. Zusätzlich bietet JavaScript die Möglichkeit asynchrone Aufrufe mithilfe des

XML-HTTP-Request Objekt zu erstellen, wodurch die Grundlage für AJAX gebildet wird.

1.2.3 JSON:

Die JavaScriptObjectNotation (JSON) ist ein kompaktes Format zum Austausch von Daten. Die Basis bildet dabei eine Teilmenge der JavaScript Programmiersprache. Trotzdem handelt es sich bei JSON um eine Vorgehensweise, die Daten sprachunabhängig auf einfache Weise strukturiert. Damit kann JSON verwendet werden, um Daten zwischen Programmen auszutauschen, die in verschiedenen Programmiersprachen implementiert wurden. Bei JSON handelt es sich um ein von Mensch und Maschine gleichermaßen zu lesendes Textformat. Der große Vorteil von liegt sowohl in der einfachen Handhabung wie auch Implementierung.

1.2.4 NodeJS:

NodeJS ist eine serverseitige Plattform in der Softwareentwicklung zum Betrieb von Netzanwendungen. Insbesondere lassen sich Webserver damit realisieren. NodeJS wird in derJavaScript-Laufzeitumgebung„V8“ ausgeführt, die ursprünglich fürGoogle Chromeentwickelt wurde, und bietet daher eine ressourcensparende Architektur, die eine besonders große Anzahl gleichzeitig bestehender Netzwerkverbindungen ermöglicht.

1.2.5 MySQL:

MySQL ist eines der weltweit verbreitetsten relationalen Datenbankverwaltungssysteme. Es ist alsOpen-Source-Software sowie als kommerzielle Enterpriseversion für verschiedene Betriebssysteme verfügbar und bildet die Grundlage für viele dynamische Webauftritte. Ein bevorzugtes Einsatzgebiet von MySQL ist die Datenspeicherung für Webservices. MySQL wird dabei häufig in Verbindung mit dem Webserver Apache und der Skriptsprache PHP eingesetzt. Viele Webdienste bedienen sich dieser Architektur und betreiben je nach Größe und Bedarf eine Vielzahl von MySQL-Servern, über die die Zugriffe aus dem Netz abgewickelt werden.

1.2.6 HTML:

HTML ist eine textbasierte Auszeichnungssprache zur Strukturierung digitaler Dokumente wie Texte mit Hyperlinks, Bildern und anderen Inhalten. HTML-Dokumente sind die Grundlage des „World Wide Web“ und werden von Webbrowsern dargestellt. Neben den vom Browser angezeigten Inhalten können HTML-Dateien zusätzliche Angaben in Form von Metainformationen enthalten, z.B. über die im Text verwendeten Sprachen, den Autor oder den zusammengefassten Inhalt des Textes.

1.2.7 AngularJS:

AngularJS ist ein clientseitiges JavaScript-Webframework zur Erstellung von Single-page-Webanwendungen nach einem Model-View-ViewModel-Muster. Die Softwareentwicklung und das Komponententesten können damit vereinfacht werden. Angular basiert auf der clientseitigen Generierung von HTML-Ansichten und Erweiterungen des Vokabulars von HTML.

1.2.8 HTTP:

Das Hypertext Transfer Protocol (HTTP) ist ein zustandsloses Protokoll zur Übertragung von Daten auf der Anwendungsschicht über ein Rechnernetz. Es wird hauptsächlich eingesetzt, um Webseiten (Hypertext-Dokumente) aus dem „World Wide Web“ in einen Webbrowser zu laden. Es ist jedoch nicht prinzipiell darauf beschränkt und auch als allgemeines Dateiübertragungsprotokoll sehr verbreitet. Will ein Nutzer Informationen an eine Webseite senden, gibt es prinzipiell zwei Methoden. Die POST-Methode überträgt Daten mit einer speziell dazu vorgesehenen Anfrageart im HTTP-Nachrichtenrumpf, so dass sie in der URL nicht sichtbar sind. Bei der GET-Methode hingegen sind die übertragenen Daten Teil der URL.

2. Implementierungssicht

Im Frontend Bereich benutzen wir *HTML* und *AngularJS*. Im Backend haben wir mittels *NodeJS* einen Server erstellt. Als Protokoll wird das für Webseiten übliche *HTTP* (Protokoll) verwendet, wobei überwiegend die *POST*-Methode bei Übertragungen seitens der Clients eingeleitet wird. Auf die *mysql* Datenbank wird bei verschiedensten Einträgen zurückgegriffen und hiermit erfolgt auch die Datenhaltung. Über den Client werden dann über die *HTTP* Methoden *JSON* Objekte verschickt, welche der Server dann verwendet.

3. Benutzersicht

Der Blog besteht aus folgenden Elementen:

- Beitrag („Post“) – Ein Beitrag kann von einem registrierten Benutzer verfasst werden.
- Gruppen („Group“) – Benutzer können geschlossene Freundesgruppen erstellen, in denen sie Beitrag verfassen können.

Grundlage der Interaktion des Benutzers mit dem Blog-Service ist die Browser-Oberfläche. Der Benutzer wird aufgefordert sich entweder einzuloggen oder bei Nichtbestehen eines Accounts sich zu registrieren. Nach gelungenem Einloggen stehen dem Benutzer folgende Funktionen zur Verfügung:

- Auszuloggen („Logout“)
- Einen Beitrag abzusetzen („Add new Post“) – Dazu werden der Titel, der Inhalt sowie die Gruppe in dem der Beitrag veröffentlicht wird, benötigt.
- Alle ihm ersichtlichen Beiträge anzuzeigen („Show Posts“)
- Sowie eine Freundesgruppe zu erstellen („Create Gruppe“).

3.1 Erstellen eines Posts

Für das Erstellen eines Posts wird der Benutzer aufgefordert den Titel, den Inhalt sowie die Gruppe in dem der Beitrag veröffentlicht wird. Wird keine Gruppe ausgewählt so wird der Beitrag in die öffentliche Gruppe gepostet. Auf die öffentliche Gruppe haben alle Benutzer Zugriff.

3.2 Anzeigen von Posts

Einem Benutzer werden stets die Posts angezeigt, zu dem er die Berechtigung dazu hat. Jeder Benutzer hat die Berechtigung die Posts der öffentlichen Gruppe und der Gruppen in denen er eingeladen ist beziehungsweise er selbst erstellt hat, anzusehen. Beim Auswählen eines Posts wird dieser in einem neuen Fenster angezeigt und zusätzlich wird dem Benutzer das Kommentieren ermöglicht.

3.3 Erstellen von Gruppen

Benutzer haben die Möglichkeit Freundesgruppen zu erstellen in denen sie private Beitrag verfassen können. Bei der Erstellung einer Gruppe werden der Name sowie die Mitglieder der Gruppe benötigt. Es kann keine Gruppe ohne Namen und/oder ohne Mitglieder erstellt werden. Für die Auswahl der Mitglieder steht dem Benutzer ein Dropdown zur Verfügung mittels dem er die Mitglieder der Gruppe auswählen kann.