

Softwaretechnikpraktikum
SS 2015
Datum: 08.06.2015

Gruppe: swp15.aae
Betreuer: Prof. Gräbe
Tutor: Klemens Schöllhorn

Projektteam:
Felix Albroscheit
Dorian Dahms
Paul Eisenhuth
Martin Lechner
Christian Seidemann
Ruth von Borell
Franz Wendt

Entwurfsbeschreibung Gesamtprojekt

Inhaltsverzeichnis

1	Allgemeines	3
1.1	Installationsanleitung	3
1.2	Backupkonzept	3
2	Produktübersicht	3
2.1	Drupal	3
2.2	Layout	4
2.2.1	Darstellung Öffentliches Profil	4
2.2.2	Darstellung Kalender	4
2.2.3	Darstellung Events	4
2.2.4	Darstellung Newsfeed	4
2.2.5	Darstellung Interaktive Karte	4
2.3	Inhalte	4
2.4	Funktionalitäten	5
2.4.1	Nutzerregistrierung	5
2.4.2	Kommentarfunktion	5
3	Grundsätzliche Struktur- und Entwurfsprinzipien	5
3.1	MVC-Modell	5
4	Struktur- und Entwurfsprinzipien einzelner Pakete	5
4.1	Datenebene	5
4.2	Darstellungsebene	5
5	Datenmodell	6
6	Testkonzept	6
6.1	Komponententests	6
6.2	Integrationstests	6

6.3	Systemtests	7
7	Glossar	7

1 Allgemeines

Die Stadtteilplattform Leipziger Osten soll mit umfangreichen Funktionen ausgestattet sein, um gute Nutzbarkeit zu gewährleisten. Akteure haben hier die Möglichkeit, ein öffentliches Profil zu erstellen, ihre Veranstaltungen einzutragen und auf einer anschaulichen Karte und einem Terminkalender mit vielen Filterfunktionen der Öffentlichkeit anzeigen zu lassen. Dadurch soll der Stadtteil Leipziger Osten attraktiver gemacht und Bausteine für Synergien einzelner Akteure gebildet werden. Diese grundlegende Struktur der Plattform wird auf Basis von Drupal realisiert. Dies ermöglicht einen einfachen Ausbau der Webseite mittels Modulen, die für Drupal zu Verfügung stehen. Neben den vorhandenen werden auch eigene Module entwickelt, beispielsweise um die benutzten Module (Kalender, Karte) mit der Datenstruktur zu verknüpfen.

Zur Strukturierung der Datensätze wird RDF benutzt und auf die bereits vorhandenen Ontologien von Leipzig Data zurückgegriffen. Um externe Daten im RDF-Format einzubinden, wird ein Konzept zur Nutzung geeigneter RDF-Ontologien entwickelt. Dadurch wird eine Grundlage für das weitere Arbeiten mit RDF und damit für die Orientierung des Projekts am Konzept Linked Open Data geschaffen. Das Projekt steht also ganz im Sinne von Open Data und bezieht damit viele Datensätze aus bereits erstellten freien Datenständen zu Akteuren der Region.

1.1 Installationsanleitung

Drupal muss auf einem Webserver installiert werden. Anweisungen hierfür sind unter <https://www.drupal.org/documentation/install> zu finden. Bei erfolgreicher Installation müssen die Drupal-Module: Kalender, Ctools, Date und View installiert werden. Es wird ein eigenständiges Modul und Thema entwickelt, so dass diese lediglich in den entsprechenden Drupal-Ordner kopiert und via Backend aktiviert werden müssen. Dabei wird die Datenstruktur (ohne Inhalte) in die Datenbank importiert. Plattforminterne Inhalte könnten über ein Datenbank-Dump erfolgen, welche als Datei im SQL-Format bereit gestellt wird. Externe Inhalte von Leipzig Open Data könnten über eine Funktion im Backend bzw. mittels täglicher CRON-Jobs in die Datenbank integriert werden.

1.2 Backupkonzept

Um ein sicheres Arbeiten zu gewährleisten, wird nicht auf der Drupalinstanz des Servers gearbeitet. Jedes Teammitglied installiert sich eine eigene Drupalinstanz auf seinem lokalen Arbeitsrechner. Der „core“ wird dabei nicht verändert, d.h. Originalthemes und -module werden nicht verändert. Sollten wir Änderungen vornehmen müssen, dann werden diese in einem abgeleiteten eigenen Modul implementiert, gleiches gilt für Themes. Die bearbeiteten Module werden über das git-Repository verwaltet und in Drupal verlinkt, so dass Fehlschläge rückgängig gemacht werden können.

2 Produktübersicht

2.1 Drupal

Im Zuge des Vorprojektes wurde bereits Drupal installiert, sowie Bausteine für die Karte und das Layout gesetzt. Für das Aufsetzen der Website mit den bereits genannten Funktionen (Kalender, Eventseite, Akteurprofil) werden im Hauptprojekt bündelweise Module implementiert, sowie die

RDF-Struktur eingebettet. Hierbei kann auf vorhandene Module zurückgegriffen werden, welche entsprechend angepasst werden bzw. über ein eigenes Modul miteinander verknüpft werden.

2.2 Layout

2.2.1 Darstellung Öffentliches Profil

Ein öffentliches Profil beinhaltet zentrale Informationen zum Akteur (Name, Adresse, E-Mail, Webseite, Telefonnummer) und eventuell Bild- und Videomaterial. Das Profil soll mit den Events, dem Kalender und der interaktiven Karte verknüpft werden. Diese soll auf dem Profil ebenso dargestellt werden. Desweiteren soll eine Kommentarfunktion für User bereitgestellt werden.

2.2.2 Darstellung Kalender

In einem Kalender werden Events zeitlich dargestellt. Über Filteroptionen können hierbei „Wunsch-Events“ angezeigt werden. Von dem Kalender aus sollen die Eventseiten und die öffentlichen Profile erreichbar sein.

2.2.3 Darstellung Events

Events werden auf einer eigenen Seite dargestellt. Diese beinhaltet Name, Datum, Ort, Zielgruppe, Sparte, die Verknüpfung zum öffentlichen Profil des Erstellers und der Darstellung des Events auf der interaktiven Karte. Die Events sollen über den Kalender, das öffentliche Profil des Erstellers und über die Übersichtskarte auf der Startseite erreichbar sein. Desweiteren soll eine Kommentarfunktion für User bereitgestellt werden.

2.2.4 Darstellung Newsfeed

Der Newsfeed soll auf der Startseite dargestellt werden und zeigt Neuigkeiten „aus dem Viertel“, der Webseite und für den Stadtteil relevante Nachrichten aus Leipzig.

2.2.5 Darstellung Interaktive Karte

Eine interaktiver Karte soll Informationen zu Akteuren und Events ortsbezogen darstellen. Die Darstellung wird mit OpenStreetmap realisiert. Die interaktive Karte sollte sowohl als Ganzes auf einer eigenen Seite präsentiert werden, als auch in entsprechenden Ausschnitten auf den Seiten von Akteuren und Events eingebunden werden. Über Filtermöglichkeiten sollen angezeigte Events aussortiert bzw. hervorgehoben werden. Die Darstellung von Ortsangaben auf der Karte sollte weitestgehend automatisch anhand von Veranstaltungs- bzw. Kontaktadressen erfolgen. Für einige Zwecke kann jedoch auch eine manuelle Darstellung durch einen Veranstalter sinnvoll sein, beispielsweise wenn eine Veranstaltung an einem Ort stattfindet, der nicht durch eine Adresse beschreibbar ist: Für bestimmte Orte innerhalb eines Parks wäre die Eingabe der exakten Koordinaten durch den Veranstalter beim Erstellen des Events die bessere Lösung.

2.3 Inhalte

Eine Datenintegration der K.I.L.O-Daten ist zur Zeit nicht möglich. Die Daten sind zum größten Teil veraltet und sollen daher nicht ausgegeben werden. Allerdings soll uns in der Zukunft eine

Excel-Tabelle übergeben werden, die interessierte Akteure enthalten soll (eventuell mit Daten, die manuell eingetragen werden müssen). Daher ist die Hauptquelle für Inhalte die Datensätze des Leipzig Open Data Projekts, welche noch über einer sinnvollen RDF-Struktur übernommen werden müssen. Inhalte von Akteuren außerhalb der übergebenen K.I.L.O.-Daten oder des Leipzig Open Datas Projekts müssen von diesen selbst eingetragen werden.

2.4 Funktionalitäten

2.4.1 Nutzerregistrierung

Ein User hat die Möglichkeit sich auf der Plattform zu registrieren, um ein öffentliches Profil zu erstellen. Von diesem aus kann er Events erstellen, welche, wie das Profil, von Besuchern der Webseite eingesehen werden können. Das Formular zur Registrierung muss von allen Seiten der Plattform aus über einen entsprechenden Link erreichbar sein, ebenso wie die Login/Logout-Funktion.

2.4.2 Kommentarfunktion

Die Kommentarfunktion soll von Besuchern und registrierten Usern genutzt werden können, um eigene Meinungen auf der Event und Profilseite zu verfassen und sich auszutauschen. Kommentare registrierter Nutzer sollen direkt veröffentlicht werden, während die von nicht registrierten Nutzern zwar gespeichert, aber noch nicht publiziert werden, um eine Vorabmoderation zu ermöglichen.

3 Grundsätzliche Struktur- und Entwurfsprinzipien

3.1 MVC-Modell

Für das Projekt adaptieren wir das MVC-Modell. Im Model werden die Datensätze verwaltet. Der Controller reagiert dann auf Nutzeranfragen, wie einer Suche und leitet diese an das Model weiter. Das Model verarbeitet diese Anfrage und gibt das Ergebnis zur Darstellung an den View weiter.

4 Struktur- und Entwurfsprinzipien einzelner Pakete

4.1 Datenebene

Die Daten werden in ein RDF-Format überführt oder stehen bereits als solches zur Verfügung. Die Strukturierung bzw. Bezeichnungen orientieren sich an den Ontologien von Leipzig Data.

4.2 Darstellungsebene

Es soll eine Hauptseite geben, die die interaktive Karte, den Newsfeed, sowie eine Übersicht über anstehende Events und Navigationsleiste beinhaltet. Weiterhin soll es Seiten geben für: Profile, Events, den Kalender, die Registrierung, die Karte, FAQ und dergleichen.

5 Datenmodell

Das Projekt orientiert sich an der RDF-Ontologie des Leipzig Data Projekts. Nach bisherigem Stand werden dabei folgende Klassen verwendet:

Für das tatsächliche Userkonto wird voraussichtlich foaf:person benutzt. Wie der Zusammenhang zwischen Account, Nutzer und Profil formuliert werden kann, ist noch zu klären. Für das Nutzerprofil war die Verwendung von foaf:PersonalProfileDocument in der Diskussion. Das Problem hiermit ist, dass damit documents gemeint sind, die die Eigenschaften einer EINZELNEN foaf:Person beschreiben, welche als foaf:maker und foaf:primaryTopic des jeweiligen PersonalProfileDocuments geführt wird. (http://xmlns.com/foaf/spec/#term_PersonalProfileDocument) Diese Spezifikation macht foaf:PersonalProfileDocument vermutlich ungeeignet für die Repräsentation von Profilen von Akteuren. Sinnvoller wäre es wohl, Akteure als foaf:Agent zu führen. Dieser Klasse kann dann ein foaf:Document (die eigentliche Profilseite) mithilfe der Eigenschaft foaf:page zugewiesen werden. (http://xmlns.com/foaf/spec/#term_page) Weiterhin soll ein Akteurprofil eventuell von mehreren Nutzern verwaltet werden können, jedoch mit verschiedenen Zugriffsrechten. Unter Verwendung der org-Ontologie zum Beschreiben von Organisationen können einer org:Organization diverse foaf:Agents, also auch foaf:Persons als „memberOf“ zugewiesen werden, damit wären also mögliche Akteursmitglieder definiert und ihnen könnten Zugriffsrechte auf das Akteursprofil zugewiesen werden. Die Sonderrechte des Akteurinhabers könnten eventuell über die Eigenschaft foaf:maker im foaf:Document definiert werden. Die eigentliche Profilseite, das foaf:Document ist als foaf:page einer org:Organization zugewiesen. Um das Profil mit Informationen zu füllen, soll dem Akteur als org:Organization über die Eigenschaft org:siteAddress eine VCard zugewiesen werden. In dieser Klasse werden die eigentlichen Informationen über den Akteur hinterlegt, beispielsweise über Eigenschaften wie vcard:hasAddress, oder vcard:hasEmail. Die VCard bietet auch die Möglichkeit, Geokoordinaten und eventuell eine Kalenderverknüpfung einzubinden.

6 Testkonzept

6.1 Komponententests

Bei der Implementierung einzelner Komponenten (Module) ist es wichtig, diese auf Funktionalität und Sicherheit zu testen. Auch wenn wir viel mit bereits vorhandenen Drupalmodulen realisieren werden, sollte jedes einzelne Modul noch einmal geprüft werden. Nur dadurch ist ein einwandfreies Funktionieren gesichert. Tests werden in Testprotokollen dokumentiert, um allen Output festzuhalten und Änderungen nachvollziehen zu können. Für die Komponententests benutzen wir das interne Drupalmodul „Testing“.

Im Rahmen des Vorprojekts wurden bisher noch keine Komponententests durchgeführt.

6.2 Integrationstests

Auch wenn alle Komponenten richtig funktionieren kann es zu Schwierigkeiten beim Zusammenwirken kommen. Es ist daher wichtig, bei dem Einfügen einer Komponente deren Zusammenspiel mit bereits implementierten Komponenten zu testen. Dies wird ebenfalls protokolliert und mittels „Testing“ geprüft.

Im Rahmen des Vorprojekts wurden bisher noch keine Integrationstests durchgeführt.

6.3 Systemtests

Nach allen Komponententests und Integrationstests muss abschließend das Gesamtsystem geprüft werden. Es wird getestet, ob das Produkt innerhalb der späteren Nutzungsumgebung funktioniert. Für immer gleiche Anfragen benutzen wir das Tool „Selenium“, welches auch bereits in dem Modul „Testing“ integriert ist. Dieses automatisiert Browseranfragen und gestattet auch Stresstests, da beliebig viele Anfragen auf einmal gestellt werden können. Auch das wird protokollarisch festgehalten.

Im Rahmen des Vorprojekts wurden bisher noch keine Systemtests durchgeführt.

7 Glossar

Leipziger Osten:

Der Leipziger Osten ist definiert durch die Stadtgebiete: Neustadt Neuschönefeld, Volkmarsdorf, Anger-Crottendorf, Sellerhausen - Stünz, Paunsdorf, Mölkau, Heiterblick, Engelsdorf, Baalsdorf und Althen-Kleinpösna. Zu klären wäre, in wie fern man Nordost und Südost mit einbezieht (oder erweiterbar macht) oder nicht.

Nutzer:

Ein Nutzer ist eine auf der Plattform registrierte Person.

Akteure:

Unter Akteure sind Folgende zu verstehen: Veranstalter, Vereine, Initiativen und Privatpersonen, die im Leipziger Osten auf verschiedene Art aktiv sind.

Veranstaltung:

Eine Veranstaltung kann von einem Akteur erstellt werden. Diese kann einmalig, regelmäßig oder unregelmäßig stattfinden.

Kurzdarstellung:

Selbstdarstellung von Akteuren auf der Stadtteilplattform, ähnlich eines Profils. Dies sollte mindestens folgende Daten umfassen: Name, Beschreibung, Adresse, sonstige Kontaktmöglichkeit (E-Mail, Facebook...), Sparte, Zielgruppe, Optional sind Bilder, ..., etc. Um ein Profil auf der Plattform anzulegen, benötigen die Akteure eine entsprechende Zugangsmöglichkeit.

Stadtteilplattform:

Eine interaktive (Online)-Plattform, welche der Organisation, Verschönerung, Attraktivität, Vermittlung, „News-Verbreitung“ und vielem mehr dienen soll. Die Plattform sollte so aufgesetzt sein, dass sie in gewisser Weise selbst funktioniert. D.h. Akteure und Kunden können sich registrieren und Programme und Angebote erstellen und aufzeigen, ohne dass alles von einem Betreiber der Seite einzeln kontrolliert werden muss. (Aus Inhaltlichen, Gesetzlichen, Datenschutz bezüglichen Gründen). Ziel der Plattform ist es, eine übersichtliche Website zu gestalten die mittels Interaktiver Karte, Kalender, etc. den Stadtteil mit seinen Akteuren attraktiv macht.