

# Arbeitsplan der Gruppe „Interaktiver Haushaltsrechner“

Verfasser:

Sebastian Zänker, Tobias Wieprich, Fabian Niehoff, Dennis Kreußel,  
Janos Borst, Wolfgang Amann

19. Januar 2015

## 1 Projektvision

Der Haushaltsrechner der Stadt Leipzig ist eine Webseite, die den Bürgerinnen und Bürgern der Stadt Leipzig seit 2007 eine Plattform im Internet zu Verfügung stellt, auf der sie sich über die aktuellen Haushaltspläne der Stadt Leipzig informieren können. Zusätzlich existierte eine Vorschlagfunktion, die es den Bürgerinnen und Bürgern ermöglichen sollte, sich in die aktuelle Haushaltsplanung einzubringen. Diese wurde jedoch 2013 eingestellt.

Ziel des Projekts „Interaktiver Haushaltsrechner für die Stadt Leipzig“ (IHR) ist es nun, diese Plattform neu zu gestalten, aufbauend auf den Anforderungsanalysen der Gruppe „haushalt-14“. Der neue Haushaltsrechner soll eine übersichtlichere Darstellung des Finanzhaushalts beinhalten und eine adäquate Vorschlagfunktion mit Diskussionsforum zur Verfügung stellen. Besonderer Fokus soll dabei auf Übersichtlichkeit und Benutzbarkeit gelegt werden, um für die Bürgerinnen und Bürger der Stadt Leipzig die Einarbeitungszeit gering zu halten und somit die Hemmschwelle zur Nutzung der Seite gering zu halten. Die Webseite soll hierzu auch barrierefrei (gemäß BITV 2.0) gestaltet sein, um ausgrenzende Wirkung zu vermeiden. Die Partizipationsfunktion soll es den Bürgerinnen und Bürgern erlauben, einfach Vorschläge zu Teilbereichen des Haushaltsplans einbringen zu können und diese mit anderen Nutzerinnen und Nutzern in einem Forum diskutieren zu können.

Die Webseite soll zudem mithilfe von Semantic Web Technologien, wie zum Beispiel RDF, verwirklicht werden, um einfache Datenintegration zu ermöglichen und den Austausch und Vergleich mit anderen anderen Haushalten, die im Semantic Web vorhanden sind, zu ermöglichen.

## 2 Voraussetzungen

Das Framework Drupal setzt verschiedene Voraussetzungen an Webserver und Datenbank. Diese müssen eingehalten werden, damit ein problemloser Ablauf garantiert ist.

## 2.1 Webserver

Drupal ist zwar ein webserverunabhängiges Framework, allerdings wird eine Installation von Apache 1.3 oder 2.x empfohlen, da Drupal mit dieser Serverart entwickelt wurde und somit mehr Erfahrungsberichte von Seiten der Community existieren. Somit kann man bei Problemen auf verschiedene Hilfestellungen in Foren zurückgreifen. Zudem sollten mindestens 60 MB Speicher für die Installation auf den Server frei sein und das Modul „mod\_rewrite“ aktiviert sein.

## 2.2 PHP

Drupal ist in PHP geschrieben, daher muss auf dem Server PHP auf dem Server mindestens Version 5.2 installiert sein, damit die aktuelle Drupal-Version 7 funktioniert.

## 2.3 Datenbank

Damit eine Datenbankanbindungen möglich ist, muss die PHP-Erweiterung für Datenbanken aktiviert sein. Drupal unterstützt von Haus aus MySQL 5 und PostgreSQL 7.4.

# 3 Designübersicht und Funktionalität

## 3.1 Design

Die Hauptaufgabe der Gruppe ist die Erstellung eines als Web-Applikation umgesetzten Haushaltsrechners für die Stadt Leipzig. Der Haushaltsrechner soll dabei möglichst intuitiv zu bedienen und übersichtlich gestaltet sein. Ein weiterer wichtiger Punkt ist auch die Barrierefreiheit, denn die Seite, auf der sich der Rechner befindet, soll auch für Menschen mit verringertem Wahrnehmungsvermögen zugänglich gemacht werden.

Die Bildfläche der Startseite wird großflächig aufgeteilt: Es gibt einen Link der die Einnahmen aufruft, einen Link für die Ausgaben, einen Link der auf eine Seite mit einer Kurzanleitung verweist und einen Link, der zu einem Diskussionsforum führt. Die Aus- und Eingaben sollen mithilfe eines interaktiven Kreisdiagramms dargestellt werden. Interaktiv heißt hierbei, dass die einzelnen Glieder des Kreisdiagramms angeklickt werden können, wodurch sich ein weiteres Diagramm öffnet, das eben jenes Segment detailliert darstellt. Durch eine solche Darstellung soll die Übersichtlichkeit der Daten gewahrt werden, denn die Menge der aufgerufenen Daten wird klein gehalten und durch das Schichtenmodell des Diagramms weiß der Benutzer zu jeder Zeit, welchem Bereich die Daten zuzuordnen sind. Sobald jedoch die Daten innerhalb des Kreisdiagramms nicht mehr übersichtlich genug differenziert werden können, werden diese unter dem Diagramm nur noch als Liste angezeigt. Die zuvor genannte Liste zeigt immer sämtliche Daten der aktuellen Abstraktionsebene an.

Die eingangs erwähnte Barrierefreiheit soll über verschiedene Mittel erreicht werden: die Farbwahl soll den existierenden Sehschwächen angepasst werden, Erklärungsvideos haben stets einen Untertitel und (wenn es möglich ist) soll die Webseite für Blinde durch entsprechende Programme aufbereitbar sein. Weiterhin soll die vorausgesetzte Einarbeitungszeit in das Thema Haushalt für den Benutzer möglichst klein gehalten werden. Das soll durch ein Mouseover-Tool geschehen, welches die Elemente dem Benutzer erklärt, sobald dieser den Mauszeiger über das jeweilige Element bewegt. Außerdem sollen Ein- und Ausgabegebiete, die durch gesetzliche

Verpflichtungen der Stadt nicht veränderbar sind, markiert werden (siehe Funktionalitäten Vorschlag ein reichen). Generell soll bei der Darstellung der Ein- und Ausgaben zwischen Ergebnisrechnung und Finanzrechnung umgeschaltet werden können.

Das Forum, welches von der Startseite aus verlinkt ist, soll nur Basisfunktionen enthalten. Es bietet eine einfache Struktur, mit der die Beiträge grob nach Themen geordnet werden.

## **3.2 Rollen**

### **3.2.1 Benutzer**

Ein Benutzer ist eine nicht weiter definierte Person, die die Seite mit dem Haushaltsrechner aufruft. Dieser startet seine Interaktion mit der Webseite im Regelfall auf der Startseite.

- ruf Daten ab:  
Der Benutzer kann über die Startseite die Daten über die Ausgaben und die Einnahmen erreichen. Dazu wird jeweils ein entsprechender Link bereitgestellt, der den Benutzer zum zugehörigen Diagramm leitet.
- registriert sich:  
Der Benutzer kann sich registrieren, damit er Vorschläge einreichen kann. Die dabei von der Website eingeforderten Daten werden nicht veröffentlicht und können nur unter vom Gesetzgeber bestimmten Bedingungen eingesehen werden.
- verfasst/verändert einen Beitrag im Forum:  
Um einen Beitrag im Forum zu verfassen, muss der Benutzer registriert sein. Um einen eigenen Beitrag zu verfassen oder zu ändern begibt er sich über den entsprechenden Link auf der Startseite in das Forum.
- reicht einen Vorschlag ein:  
Auch hierfür muss der Benutzer registriert sein. Um einen Vorschlag einzubringen gibt es zwei Wege: Einerseits kann man dazu ein leeres Formular anfordern und es direkt ausfüllen. Andererseits kann man in den Einnahmen- und Ausgabenbereich navigieren und ein teilweise vorausgefülltes Formular anfordern.

### **3.2.2 Administrator**

Ein Administrator ist ein Benutzer mit zusätzlichen Rechten, die die Verwaltung der Website betreffen.

### **3.2.3 Moderator**

Eine Person die mit der Überwachung der Forumsaktivitäten beauftragt ist. Sie kann Beiträge verändern, verschieben, löschen und bei groben Verstößen gegen die Forumsordnung den betreffenden Benutzer sperren.

### 3.3 Funktionalitäten

#### 3.3.1 Muss-Funktionalitäten

- Darstellung des Finanzhaushalts Leipzig:  
Die Daten (sowohl Einnahmen als auch Ausgaben) müssen als interaktives Kreisdiagramm mit aufrufbaren Segmenten dargestellt werden können. Bei jedem Aufruf eines Segments wird ein neues Diagramm erstellt, dessen Segmente die Bestandteile des aufgerufenen Segments sind.
- Forum für Diskussionen:  
Die Plattform muss Möglichkeiten zur Integration eines einfachen Forums beinhalten.
- Vorschläge einbringen:  
Es muss ein Formular bereitgestellt werden, mit dem Benutzer Vorschläge einbringen können. Diese berücksichtigen auch Ausgabegebiete, die nicht verändert werden können, sodass der Benutzer an entsprechender Stelle einen entsprechenden Hinweis angezeigt bekommt.
- Vorschläge bewerten:  
Vorschläge müssen für andere Benutzer einsehbar sein und von diesen bewertet werden können.
- Vote-Charts erstellen:  
Vorschläge sollen nach deren Bewertung sortiert werden können.
- Vorschläge einordnen:  
Es soll eine grobe Ordnungsstruktur vorhanden sein, die die Vorschläge auf der obersten Abstraktionsebene des Haushaltsrechners Themen zuordnet.
- Vorschlagszeitraum soll beachtet werden:  
Es gibt einen Zeitraum, in dem eingehende Vorschläge bearbeitet werden können. Versucht ein Benutzer einen Vorschlag außerhalb dieser Zeit einzureichen, wird er auf diesen Umstand von der Plattform hingewiesen.

#### 3.3.2 Kann-Funktionalitäten

- Bereitstellung von Erklärvideos:  
Der Benutzer ist in der Lage sich Erklärungsvideos anzusehen, um die Verständlichkeit der Erklärung der Seite (siehe oben) zu erhöhen und die Einarbeitungszeit zu verringern.
- Sprachausgabe ändern:  
Damit Benutzer, die nicht Deutsch sprechen, sich besser zurecht finden, kann die Seite auch in englischer Sprache aufgerufen werden.
- Hauptthemen:  
Die im Forum innerhalb eines bestimmten Zeitraums am meisten diskutierten Beiträge werden auf die Startseite verlinkt.

## **4 Arbeitspakete**

### **4.1 Arbeitsumgebung (8%)**

In unserem ersten Arbeitspaket werden wir uns mit den Grundlagen zur Umsetzung des Projekts beschäftigen. Auf dem Praktikumsserver wird Drupal sowie eine Testumgebung dafür aufgesetzt. Danach machen wir uns mit Drupal vertraut und beginnen damit, die Kernfunktionen zu testen, die im späteren Verlauf des Projektes benötigt werden. Desweiteren wird noch das Basiswissen für RDF vertieft, eine Datenbank zum testen eingerichtet und sich in die Grundlagen von SPARQL eingelesen.

### **4.2 Prototypische Testseite (10%)**

Nachdem die Grundlagen und Kernfunktionen erlernt wurden, kommt das erste essentielle Projekt: Es wird eine kleine Testseite entwickelt, die bereits die Daten aus den RDF ein- und auslesen kann.

### **4.3 Implementation der Informationsebene (40%)**

In diesem Arbeitsschritt findet der größte Arbeitsabschnitt statt. Zu Beginn wird ein Designkonzept entwickelt, welches die exakten Designanforderungen an die spätere Website enthält. Nach dessen Umsetzung das Designkonzept fertig ist und wir uns für spezifische Bibliotheken zur Datenvisualisierung entschieden haben, wird an der Visualisierung der Daten nach den Vorgaben des Designkonzepts gearbeitet.

### **4.4 Testphase I (10%)**

In diesem Abschnitt wird unsere Informationsebene gründlich geprüft. Es werden Fehler, Inkonsistenzen und weitere Probleme gesucht und behoben.

### **4.5 Forum und Partizipation (20%)**

Zu Beginn dieses Arbeitspaketes wird zunächst ein Basis-Forum aufgebaut, welches die zugehörigen Usergruppen besitzt. Dannach wird die Verbindung zur Informationsebene aufgebaut (Vorschläge → Threads im Forum etc.). Nachdem dieses funktioniert wird an den Muss-Funktionen für das Forum gearbeitet. Das heißt, dass wir zunächst konkret Metadaten eintragen, eine Registrierung einbauen, einen Abschnitt für Vorschläge über den Haushalt einbauen und abschließend eine Bewertungsfunktion einbauen.

### **4.6 Testphase II (5%)**

Hier werden die Funktionen aus dem 4. Arbeitspaket wie in der ersten Testphase geprüft.

### **4.7 Handbuch (2%)**

Im letzten Arbeitspaket wird das Benutzerhandbuch verfasst, um die Nutzung des interaktiven Haushaltsrechners allen Bürgern der Stadt Leipzig zu ermöglichen.

## 4.8 Optionales (unspezifiziert, führt zu >100%)

Hier werden die von uns als optional angesehen Features eingebaut. Als optional sehen wir: Eine detaillierte Suchfunktion, sowie andere Sortierfunktionen für das Forum, die Möglichkeit Themenvorschläge an die Redaktion (offizielle Diskussionen) zu senden, Guides und Ähnliches zur Unterstützung der Seitennutzung für weniger technik-affine Bevölkerungsgruppen und einen selbst zusammenstellbaren Haushaltsrechner mit Hilfe von Schiebereglern zusammenstellen zu können. Abschließend werden die jeweilig eingebundenen optionalen Punkte vor Übergabe getestet.

## 5 Qualitätssicherung

Um ein zufriedenstellendes Endergebnis des „Interaktiver Haushaltsrechner für die Stadt Leipzig“ gewährleisten zu können, sind vor allem Qualität und Benutzbarkeit des Endprodukts wichtig.

### 5.1 Übersicht der Qualitätsanforderungen

	Sehr gut	Gut	Normal	Nicht relevant
Produktqualität	X			
Funktion		X		
Zuverlässigkeit		X		
Benutzbarkeit	X			
Effizienz			X	
Änderbarkeit		X		
Übertragbarkeit			X	

### 5.2 Nichtfunktionale Anforderungen

#### 5.2.1 Benutzbarkeit

Vom Endprodukt soll vor allem eine intuitive Bedienung gewährleistet werden. Im Bereich Design und Navigation sollte man sich hierfür an den Bundeshaushaltsrechner bzw. bereits bestehende Haushaltsrechner anderer Städte orientieren.

#### 5.2.2 Änderbarkeit

Da einige Funktionalitäten des Partizipationsteils zum Haushaltsrechner vorerst als optional erklärt wurden spielt eine angemessene Änderbarkeit eine große Rolle. Daneben liegen nach Abschluss des Projekts die Wartung und Pflege des Produkts nicht in unserem Aufgabenfeld. Damit dies also effizient erfolgen kann, ist besonders darauf zu achten, allgemeine Standards zur Modellierung sowie Coding- und Dokumentationsstandards einzuhalten. Nähere Informationen hierzu finden sich im Qualitätssicherungskonzept.

#### 5.2.3 Übertragbarkeit

Da das Projekt nach unserem Beitrag noch nicht vollständig abgeschlossen sein wird oder von der Stadtverwaltung administriert werden wird, ist die Formatierung sowie das Versehen des Codes mit Kommentaren von großer Bedeutung. Hierbei werden wir uns stark an die Drupal-Coding Standards anlehnen.

## 6 Glossar

**Drupal** ist ein freies (GPL) und kostenloses Content-Management-Framework auf der Basis von PHP. Es besteht dabei aus einem Core, der aus anderen Content-Management-Systemen bekannte Funktionen liefert und Modulen, mit denen man zusätzliche Funktionen implementieren kann.

**JavaScript** ist eine Skriptsprache, die vor allem dafür benutzt wird, um Skripten auf Client-Seite zu erlauben, mit dem Benutzer zu interagieren, den Browser zu kontrollieren, asynchron zu kommunizieren und den Dokumenteninhalte, der angezeigt wird, zu verändern. Andere Verwendungen findet JavaScript z. B. auf Server-Seite innerhalb von Frameworks wie Node.js oder bei der Entwicklung von Anwendungen für mobile und Desktop-Endgeräte.

**OWL** Die Web Ontology Language dient zur Spezifikation von Beschreibungssprachen zur Erstellung von Ontologien für das semantische Web. Dabei basiert OWL auf der RDF-Syntax, geht aber in seiner Mächtigkeit teilweise deutlich darüber hinaus.

**RDF** Das Resource Description Framework als Konzept erweitert die Möglichkeit des Webs, Inhalte miteinander zu verbinden. Hierbei wird versucht, nach semantischen Zusammenhängen zu ordnen. Die Aussagen die die eigentlichen Daten in RDF sind, betreffen sogenannte Ressourcen. Diese Ressourcen sind eindeutig bezeichnete Dinge der Welt, die durch Tripel oder Graphen in eine Relation mit anderen Ressourcen gebracht werden. Nicht nur durch das mit RDF im Zusammenhang benutzten Vokabular ähnelt dieses Modell stark an die klassischen Modelle wie UML-Diagramme oder Entity-Relationship-Modelle, auch die Modellierungsweisen sind stark miteinander verknüpft, wodurch RDF als Grundansatz des Semantischen Webs an Bedeutung gewinnt.

**SPARQL** (rekursives Akronym für *SPARQL Protocol And RDF Query Language*) ist eine Abfragesprache für RDF, d. h. eine semantische Abfragesprache für Datenbanken, die dazu in der Lage ist, Daten, die im RDF-Format gespeichert sind, aufzurufen und zu manipulieren.

**Semantic Web** Im Rahmen zur Weiterentwicklung zum Internet der Dinge ist es nötig, dass von Menschen zusammengetragene Information von Maschinen verarbeitbar gemacht wird. Aus diesem Grund sollen nach dem Ansatz des semantischen Web (*Semantic Web*) in menschlicher Sprache ausgedrückten Informationen mit einer eindeutigen Beschreibung versehen werden, die von Computern verarbeitbar und optimalerweise auch verstanden werden kann.

**Triplestore** Ein Triplestore ist eine Datenbank zur Speicherung und Abfrage von Tripeln durch semantische Abfragen. Ein Daten-Tripel besteht aus Subjekt, Prädikat und Objekt, z.B. „Franz kennt Fritz“.

**Web-Anwendung** Eine Webanwendung läuft im Allgemeinen auf einem Webserver, kann aber auch auf verschiedene Anwendungsserver ausgelagert sein. Hierbei ruft der Benutzer die Anwendung auf, indem er z. B. in einem Browser die URL des Webserver eingibt. Der Browser ist hierbei die Präsentation der vom Server gestellten Antworten auf die Anfragen des Benutzers.

Dabei kann man grundsätzlich zwischen der Standalone und der integrierten Architektur unterscheiden. Bei ersterer ist die Anwendung ein eigenständiges Programm, welches für jede Anfrage eines Nutzers neu ausgeführt werden muss. Bei zweiterer ist die Anwendung entweder

direkt Teil des Servers oder zu interpretierenden Skriptes, wodurch man das Problem, immer wieder erneut ein Programm für jede Anfrage starten zu müssen, umgeht.

## 7 Quellenverzeichnis

Letzter Aufruf von allen Internet-Quellen am 19. Januar 2015.

- <http://de.wikipedia.org/wiki/Drupal>
- <http://en.wikipedia.org/wiki/JavaScript>
- [http://en.wikipedia.org/wiki/Resource\\_Description\\_Framework](http://en.wikipedia.org/wiki/Resource_Description_Framework)
- <http://en.wikipedia.org/wiki/SPARQL>
- [http://de.wikipedia.org/wiki/Semantisches\\_Web](http://de.wikipedia.org/wiki/Semantisches_Web)
- <http://en.wikipedia.org/wiki/Triplestore>
- [http://en.wikipedia.org/wiki/Web\\_application](http://en.wikipedia.org/wiki/Web_application)
- [http://en.wikipedia.org/wiki/Web\\_Ontology\\_Language](http://en.wikipedia.org/wiki/Web_Ontology_Language)