

**Heinrich-Heine-Universität Düsseldorf**  
**Philosophische Fakultät**  
**Institut für Sprache und Information**  
**Abteilung für**  
**Informationswissenschaft**



**Haydar Akyürek, Christian Born,  
Valerie Claessen, Regina Stodden, Sabrina Wirkner**

# **Easy Living**

## **Projektbericht**

<b>Abgabefrist:</b>	<b>30.09.2017</b>
<b>Abgabedatum:</b>	<b>30.09.2017</b>
<b>Dozenten:</b>	<b>Dr. Tobias Siebenlist, Univ.-Prof. Dr. Wolfgang G. Stock</b>

# Inhaltsverzeichnis

Inhaltsverzeichnis .....	II
Abbildungsverzeichnis .....	III
Grundidee (Valerie Claessen).....	4
Usecases .....	6
Konzept (Haydar Akyürek) .....	7
Verwendete Daten .....	7
Probleme bei der Verwendung der Daten .....	8
Verwendete Technologien .....	9
Ablauf (Regina Stodden) .....	11
Aufgabenverteilung und Projektmanagement .....	11
Meilensteine .....	11
Meilenstein 1 .....	12
Meilenstein 2 .....	12
Meilenstein 3 .....	13
Meilenstein 4 (Abschlusspräsentation) .....	14
Ergebnis (Christian Born).....	15
Suche & Navigation .....	15
Karte .....	17
Filter & Open Data .....	18
Ausblick (Sabrina Wirkner).....	21
Einsatzgebiet .....	21
Erweiterungsmöglichkeiten.....	21
Reflexion und Kritik .....	23
Fazit (Christian Born).....	25
Literaturverzeichnis .....	26

# Abbildungsverzeichnis

Abbildung 1: Suchmaske Immobilienscout24.....	4
Abbildung 2: Selbst inserierte Lagebeschreibung einer Immobilie. ....	4
Abbildung 3: Ausschnitt aus dem Wohnwert-Barometer der TU Darmstadt, Thema “Behaglichkeit” .....	5
Abbildung 4: Seite beim initialen Aufruf .....	15
Abbildung 5: Suche nach “Köln” und Ergebnis.....	16
Abbildung 6: Hierarchische Navigation. ....	17
Abbildung 7: Filter-Dropdowns und Auswahl .....	18
Abbildung 8: Schnittflächenberechnung und -anzeige.....	19
Abbildung 9: Open Data Schieberegler.....	20

## Grundidee (Valerie Claessen)

Bisher dienen zur Suche nach einem gewünschten Wohnort entweder Suchplattformen wie Google Maps, die rein über die Suche nach einem Ortsnamen oder der Orientierung auf einer Karte funktionieren, oder dem Immobilienmarkt und damit verbundenen Plattformen wie beispielsweise Immobilienscout24<sup>1</sup>, die Immobilien nach Ort und Preisrahmen sortiert anzeigen. Andere verwendbare und durchaus relevante Daten, die das Umfeld eines potentiellen Wohnortes betreffen, können so jedoch nicht in die Suche miteinbezogen werden, sind jedoch für den Nutzer oft von primärer Bedeutung.

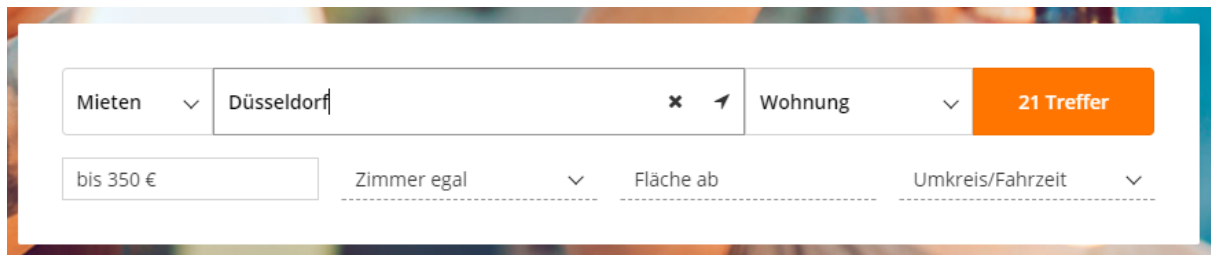


Abbildung 1: Suchmaske Immobilienscout24

Möchte ein Wohnungssuchender beispielsweise eine Wohnung finden, die sich besonders nah an einem Kindergarten befindet, kann er dies nur herausfinden, indem er dies für jede vorgeschlagene Wohnung einzeln nachrecherchiert. Es besteht keine Möglichkeit dieses Kriterium direkt in die Suche miteinzubeziehen. Oftmals werden von Vermietern oder Verkäufern zwar Informationen auf der jeweiligen Seite des Objekts aufgezeigt, jedoch sind diese weder in ihrer Validität noch in ihrer Relevanz für den Nutzer geprüft.

### Lage

Das Objekt liegt in Düsseldorf Benrath. Die benrather Innenstadt ist in weingen Gehminuten zu erreichen. Die Bus-und bahnverbindungen sind hervorragend.

Abbildung 2: Selbst inserierte Lagebeschreibung einer Immobilie.

---

<sup>1</sup> <https://www.immobilienscout24.de/>

Um dieser Problematik entgegen zu wirken, wurde die Plattform *EasyLiving* implementiert, welche das Ziel verfolgt, den idealen Wohnort anhand für den Nutzer relevanter Kriterien zu bestimmen. Dem Nutzer soll so ermöglicht werden, verschiedenste Faktoren für seine Wohnungssuche genau vorzubestimmen. Ähnlich zu herkömmlichen Immobilienseiten kann zunächst der Ort, in dem sich eine Immobilie befinden soll, festgelegt werden. Zusätzlich können Kriterien (Filter) ausgewählt werden, die sich entweder in einem bestimmten Radius in der Nähe des Objekts befinden sollen oder einen bestimmten Mindestabstand davon erfüllen müssen. Informationen, die das ausgewählte Gebiet betreffen, können zusätzlich angezeigt werden und so ergänzend bei der Auswahl behilflich sein.

Um einen umfassenden und für den Nutzer relevanten Kriterienkatalog zu erstellen, wurde das Wohnwert-Barometer (Hegger et al., 2010) der technischen Universität Darmstadt herangezogen, welches für Anwohner wichtige Merkmale der Umwelt aufzählt, die diese entweder auf- oder abwerten. Daraus wurde eine Auswahl an Themen getroffen, die die Kriterien von *EasyLiving* bilden. Unter dem Thema *Behaglichkeit* findet sich beispielsweise der Punkt *akustische Behaglichkeit* (Abb. 3), der über die Anzeige des Lärmpegels umgesetzt wurde. Das Thema *Versorgung* schließt die Punkte *Verkehrsanbindung* und Abdeckung von Kindergärten, Schulen und Naherholungsflächen mit ein, die in Form von wählbaren Kriterien eingebunden wurden.

<b>K01 Behaglichkeit</b>	1	Akustische Behaglichkeit	1a	Schallschutzanforderungen Wohnung u. akustische Zonierung
			1b	Schallschutzanforderungen zwischen Wohneinheiten
	2	Visuelle Behaglichkeit	2a	Natürliche Belichtung Wohnung
			2b	Belichtung und Beleuchtung Erschließungsflächen
	3	Thermische Behaglichkeit	3a	Thermischer Komfort Sommer
			3b	Thermischer Komfort Winter
	4	Raumluftqualität	4a	Schadstoffe in Materialien, Gering emittierende Materialien
			4b	Kontrollierte Frischluftzufuhr
	5	Sicherheit	5a	Sicherheit Aussenraum
			5b	Sicherheit Gebäude

Abbildung 3: Ausschnitt aus dem Wohnwert-Barometer der TU Darmstadt, Thema “Behaglichkeit”

*EasyLiving* bietet so, im Gegensatz zu anderen Plattformen, ein an die Bedürfnisse des Nutzers angepasstes Suchinterface, in welchem sich durch verschiedene Filter zunächst eine Vorauswahl möglicher Wohnorte treffen lässt. Bei Verfeinerung der Suche sind alle Einstellungen flexibel skalierbar, was verschiedenste Anwendungsmöglichkeiten für unterschiedliche Zielgruppen bietet.

## Usecases

Die Skalierbarkeit der Plattform ermöglicht verschiedenste Suchanlässe, die anhand einiger Usecases aufgezeigt werden:

1. Einfache Ortssuche: Über die Ortssuche kann der Nutzer auf verschiedenen Wegen zunächst nach einem ihm bekannten Ort suchen und diesen auf der Karte von EasyLiving anzeigen lassen. So können Nutzer, denen bereits bekannt ist, in welchem Ort sie einen Wohnort suchen, leicht ihr Suchgebiet einschränken.
2. Suche über die Postleitzahl: Ähnlich wie bei der Ortssuche kann der Nutzer auch eine ihm bekannte Postleitzahl eingeben und wird so zum gewünschten Kartenausschnitt geleitet.
3. Freie Navigation auf der Karte: Möchte ein Nutzer einfach frei über die Karte navigieren, um so einen attraktiven Wohnort zu finden, kann er dies über das einfache Scrollen über die Karte. Mit einem Rechtsklick kann der so gefundene Ort ausgewählt und für die weitere Suche verwendet werden.
4. Suche nach Ortsteilen: Möchte der Nutzer einen bestimmten Bereich eines Ortes finden, so kann er diesen auch schlicht eingeben. Kommt es zu Überschneidungen mit anderen Orten, so werden dem Nutzer alle passenden Treffer angezeigt.  
Beispiel: Suche nach *Altstadt*, wobei der Nutzer die Düsseldorfer Altstadt meint. Hier werden alle Orte auf der Karte markiert, die dem Suchterm entsprechen, und zusätzlich noch als Textauswahl angezeigt.

# Konzept (Haydar Akyürek)

## Verwendete Daten

Das OpenStreetMap Projekt (OSM) ist ein freies Web-Projekt, welches durch kollaborative Beteiligung seiner Nutzer eine möglichst genaue und detaillierte Weltkarte abbilden soll. Nutzer können in der OSM-Karte, neben allgemeiner Informationen wie Stadt- und Straßennamen, sowie Postleitzahlen geographischer Grenzen, auch Orte mit speziellen Tags, den sogenannte *Amenities* (englisch für Annehmlichkeit) markieren. Aktuell existieren über 230 einzigartige Tags<sup>2</sup>. Darunter diverse spezifische Tags für öffentliche Einrichtungen, Gesundheitswesen, Freizeitaktivitäten uvm.

OpenData ergänzen die bereits vorhandenen OSM-Daten in diversen Themengebieten. OpenData sind, wie der Name vermuten lässt, offene Datensätze, die von öffentlichen und privaten Institutionen zur Verfügung gestellt wurden und prinzipiell von jedem genutzt werden können. Im Gegensatz zu den in der Regel allgemeinen Informationen der OpenStreetMap Daten sind die OpenData Datensätze konkreter und meist auf gewisse Themengebiete beschränkt. Die Stadt Köln bietet mit ihrer OpenData Plattform<sup>3</sup> ein exemplarisches Beispiel für eine gute Umsetzung der OpenData Idee.

Für dieses prototypische Projekt wird, primär aus Ressourcengründen, lediglich der OSM-Datensatz von Nordrhein-Westfalen verwendet. Aus den OSM-Datensätzen werden sechs, für die Wohnqualität wichtige, Kategorien bestimmt und mit passenden Tags befüllt, damit diese Tags später in die Suche einfließen können. Die Kategorien sind:

- Bildung
- Freizeit
- Geschäfte
- Gesundheit
- Religion
- Transport

---

<sup>2</sup> <http://wiki.openstreetmap.org/wiki/DE:Key:amenity>

<sup>3</sup> <https://www.offenedaten-koeln.de/>

Zusätzlich werden OpenData für die Stadt Köln mit in das Projekt einfließen. Ergänzend zu den genannten Kategorien lassen sich bei einer Suche innerhalb Kölns noch folgende Informationen anzeigen:

- Durchschnittliches Alter und Anteil an (Jugend-) Arbeitslosigkeit
- Wahlergebnisse der letzten Landtagswahlen
- Durchschnittliche Mietpreise
- Lärmpegelmessungen
- LKW-Verbotzonen

Diese Datensätze liegen im OSM-Datensatz nicht vor, enthalten jedoch nützliche Zusatzinformationen über die Orte. Alle Datensätze, bis auf die LKW-Verbotzonen, liegen für jeden Kölner Stadtteil separat vor.

Durch die Kombination der OSM und OpenData Daten wird versucht, wichtige Informationen über die Wohnqualität von Ortschaften zu geben und aussagekräftige Filterkriterien und Resultate zu liefern. Es werden bewusst nur die OpenData der Stadt Köln verwendet, da es als Idealbeispiel für die Umsetzung der Projektidee dienen soll. Da keine flächendeckend verfügbaren OpenData vorliegen, ist die Projektidee nicht für den gesamten Kartenausschnitt Nordrhein-Westfalen umsetzbar.

### **Probleme bei der Verwendung der Daten**

Bei der Datenevaluation fiel auf, dass die Qualität der OSM-Datensätze von Stadt zu Stadt stark abweicht. Die größten Probleme waren lückenhafte Datensätze, sowie falsche bzw. mehrfache Einträge zu einem Objekt. Das ist ein bekanntes Problem, wenn Datensätze kollaborativ und ohne strenge Überprüfung erweitert werden. Zudem sind die Datensätze in größeren Städten wie Düsseldorf oder Köln deutlich genauer und umfangreicher als in kleineren Städten und Dörfern. Nahezu alle durch die unvollständigen Daten entstandenen Fehler wurden in der Umsetzung speziell behandelt: Einzig die mehrfachen und falschen Einträge können vom System nicht überprüft werden und müssen so akzeptiert werden.

Auch in den OpenData gab es ähnliche Probleme. Einerseits war auffällig, dass sensible, ortsbezogene Daten wie Kriminalitätsraten und Anteile der Menschen mit Migrationshintergrund nicht öffentlich verfügbar waren. Der Grund dafür ist jedoch ersichtlich. Solche



Informationen können, gerade im Bereich Immobilien und Wohnungssuche, ausschlaggebend für Nutzer sein. Andere Daten wiederum sind nicht ausreichend dokumentiert und können daher nicht ohne Weiteres verwendet werden.

Außerdem gibt es, wie bereits erwähnt, kaum flächendeckende OpenData. Köln ist eine der wenigen Städte in NRW, die umfangreich OpenData anbieten. Die meisten anderen Städte hatten die oben aufgeführten OpenData Datensätze nicht. Diese lediglich ortsbezogenen OpenData führen dazu, dass diese Projektidee mit einem größeren Kartenausschnitt vorerst nicht umsetzbar ist.

## **Verwendete Technologien**

Zur Umsetzung des Projektes müssen geeignete Technologien und Tools verwendet werden, primär um alle Informationen korrekt verarbeiten zu können. Eine Kombination verschiedener Technologien machte die Umsetzung erst möglich. PostgreSQL wurde als Datenbanktechnologie gewählt, da die zusätzliche PostGIS-Erweiterung das Speichern und Abfragen von geographischen Informationen und Objekten, wie bspw. die Ermittlung der Distanz zweier Punkte oder die Schnittmenge zweier Polygone ermöglicht.

Das eigentliche Programm wurde in der Programmiersprache Python mit Hilfe des Django Web-Frameworks umgesetzt. Django setzt auf das Model-View-Template (MVT) Entwurfsmuster und gibt eine strikte Trennung der Programmbestandteile vor. Die Template-Komponente dient zur Darstellung der Webseite und hat keine steuernde Logik. Die Models enthalten die Daten und Objekte des Programms, welche in der Template-Komponente visualisiert und durch die Bedienungselemente des Templates verändert werden können. Die View wirkt als Bindeglied zwischen den beiden Bestandteilen, steuert die Datenkommunikation zwischen Model und Template und sorgt dafür, dass dem Template die richtigen Informationen vorliegen, damit diese dargestellt werden können. Ergänzend zu Django und PostgreSQL kommt das Bootstrap Framework und die JavaScript Bibliothek Leaflet zum Einsatz. Bootstrap ermöglicht eine umfangreichere Gestaltung der Darstellung und mit Leaflet ist eine Karten-Visualisierung und -Manipulation möglich.

Mit dem Open-Source Tool TileMill werden eigene Karten, sogenannte Tiles, generiert, um diese anschließend durch Leaflet verwenden zu können. Mit TileMill lassen sich Karten und Kartenelemente nach eigenen Vorgaben generieren, formatieren und auf dem

Webserver als Bildgrafiken abspeichern. Das hat den entscheidenden Vorteil, dass keine externen Quellen für Tiles verwendet werden müssen und somit keine Abhängigkeit zu anderen Diensten besteht.

## Ablauf (Regina Stodden)

Nach der Ausarbeitung der Grundidee und des Konzeptes wurden vier Meilensteine angesetzt und die Aufgaben grundlegend verteilt.

### Aufgabenverteilung und Projektmanagement

Um diese Meilensteine zu erreichen, wurden wöchentliche Treffen angesetzt, in denen ein Austausch über den aktuellen Stand, Besprechungen von Problemen, Treffen von wichtigen Entscheidungen und weitere Planungen möglich waren. Zudem wurden die Termine dafür genutzt, für die nächste Woche(n) Aufgaben zu verteilen, sowie Fristen zu setzen. Neben den persönlichen Treffen wurden eine Facebook-Gruppe, ein Google-Drive und ein GitHub-Repository erstellt, um einen dauerhaft abrufbaren Überblick über die Entwicklung des Projektes und die anstehenden Aufgaben zu ermöglichen.

Für die grundlegende Verteilung wurde geplant, dass Christian, Haydar, Kai und Regina die Programmierung übernehmen, wobei Kai sich speziell auf JavaScript und Haydar auf die Integration und Aufbereitung der Datenbanken konzentrieren sollte. Christian und Regina sollten sich zudem mit OpenData beschäftigen. Das Design, Frontend und Präsentieren der Meilensteine sollte von Sabrina und Valerie übernommen werden.

Wie die folgende Beschreibung des Verlaufes des Projektes zeigt, wurde die Einteilung größtenteils eingehalten. Jedoch wurden auftretende Probleme teilweise durch Anregungen, Hilfestellungen oder Übernahme von Teilaufgaben von anderen Mitgliedern oder dem gesamten Team behoben, sodass die Grenzen der Aufgabenverteilung verschwimmen.

### Meilensteine

Laut der Planung sollten bis zum ersten Meilenstein (13. Juni, Dauer vier Wochen) die ersten Grundfunktionen implementiert werden. Im zweiten Meilenstein (4. Juli, Dauer drei Wochen) sollten diese erweitert werden, damit in den darauffolgenden drei Wochen ein Prototyp entwickelt werden kann, der am dritten Meilenstein (25. Juli) präsentiert

werden sollte. In einem vierten Meilenstein (kein festes Datum) wurde geplant, dass der Prototyp mit Zusatzfunktionen angereichert werden kann.

## **Meilenstein 1**

Für den ersten Meilenstein wurden zuerst die OpenStreetMap Datenbank und die Software lokal auf den Rechnern eingerichtet. Die Einrichtung auf dem zur Verfügung gestellten Server gestaltete sich schwieriger und konnte zu diesem Zeitpunkt nicht umgesetzt werden (Kai). Neben der Erstellung der Projektstruktur und einer ersten Version des Frontends (Valerie), wurde das EasyLiving Logo designt (Sabrina), OpenData gesichtet sowie ausgewählt (Christian und Regina) und die OpenStreetMap Datenbank integriert. Bei der OpenStreetMap Datenbank traten erste Probleme bezüglich der Postleitzahlen auf, die aber durch eine neue default.style-Datei behoben werden konnten (Haydar), so dass erste Suchanfragen implementiert werden konnten (Haydar).

Um die Suchanfragen im Frontend verwenden zu können, musste die Karte darin integriert werden (Kai). Dies wurde kurzfristig implementiert (Kai und Regina), da die Umsetzung bzw. die Einhaltung der Frist wegen fehlender lokaler Einrichtung nicht möglich war (Kai).

## **Meilenstein 2**

Nach Erreichen des ersten Meilensteins wurden als nächstes die Suchanfragen auf Stadt-, Stadtbezirk-, Stadtteil-, und Postleitzahl-Suche erweitert (Haydar). Das Frontend wurde für die Auswahl der OpenStreetMap Daten, wofür zuvor eine Vorauswahl getroffen wurde (Sabrina), mit Schieberegler und Dropdown Menüs erweitert (Valerie). Für die Filter der OpenData wurden die ausgewählten Datensätze in eine Datenbank integriert (Christian) und passende Abfragen dazu und zu den OSM-Daten implementiert (Haydar). Da die Anbindung der Daten an die Karte mittels eines AJAX-Calls erst während der Präsentation fertiggestellt wurde (Kai), wurde stattdessen eine Notlösung vorgestellt (Regina). Zudem wurden für die Karte eine erste Version der Umkreissuche implementiert (Regina) und Kartenausschnitte bzw. Tiles farblich designt (Sabrina). Der Server konnte erneut nicht eingerichtet werden (Kai).

### Meilenstein 3

Bei der nächsten Meilensteinpräsentation, die als Abschlusspräsentation geplant war, musste umstrukturiert werden, da ein Gruppenmitglied die besprochenen Fristen und gemachten Versprechungen wiederholt nicht eingehalten hatte und somit das Team verlassen musste (Kai). Trotz Umverteilung der nun offenen Aufgaben wurde befürchtet, den Zeitplan nicht einhalten zu können, weswegen die Abschlusspräsentation um drei Wochen verschoben und zu dem zuvor geplanten Zeitpunkt eine Zwischenpräsentation angesetzt wurde. Die Einrichtung des Servers wurde wegen Zeitmangel nicht weiter berücksichtigt, da die Erarbeitung eines Prototypen für wichtiger erachtet wurde.

Bis zu der Zwischenpräsentation wurden weitere OpenData integriert (Christian) und Methoden für deren Verwendung im Frontend implementiert und dauerhaft weiterentwickelt (Haydar). Weitere Methoden wurden für die Filtersuche, -überschneidung, -anzeige ebenfalls implementiert und dauerhaft weiterentwickelt (Regina). Die Berechnung der routenbasierten Entfernung von Datenpunkten mit pgRouting scheiterte (Regina), sodass stattdessen eine Einschränkung der potentiellen Wohnflächen mit der Art der Nutzungsfläche getroffen wurde (Regina). Des Weiteren wurde, trotz zwischenzeitlichen Problemen mit der hierarchischen Suche bzw. der uneinheitlichen Verwendung des Admin-Levels in OpenStreetMap, diese Suche im Backend (Haydar) und Frontend (Christian) implementiert. Für die Anzeige, das Auslesen und die Zwischenspeicherung dieser Daten wurde die Verwendung von Sessions integriert (Christian).

Das Frontend wurde aufgrund hinzukommender Elemente und daraus resultierendem Platzmangel grundlegend umgestaltet (Sabrina und Valerie). Weitere Änderungen sind zudem der Zeitpunkt der Einblendung der Anzeige von Elementen für eine bessere Nutzerführung (Valerie), Einbindung der erstellten Tiles und Integration der OSM Kategorien (Sabrina).

Die Nutzerinteraktion wurde des Weiteren durch Fehlerbehebungen (alle), Änderung von Fehlermeldungen (Sabrina und Valerie) und Anzeige, sowie Löschoption der Filter (Valerie) optimiert. Die Anzeige der Karte und der Umfang derer Elemente wurde ebenfalls erweitert, indem die Farben der Stadtteilpolygone und Anzeige der Namen auf einem Tooltip dynamisch angepasst (Christian), individuelle Marker (Sabrina) hinzugefügt, geclustert (Regina) und mit Hover-, sowie Popup-Effekten ausgestattet (Christian und Regina) und die Anzeige der OpenData integriert wurde (Regina).

**Meilenstein 4 (Abschlusspräsentation)**

Für die abschließende Präsentation wurden größtenteils neue Ideen eingearbeitet und Fehler ausgebessert (Christian und Sabrina), wie das Löschverhalten von Elementen der Karte, sowie das Zoomverhalten bei neuen Nutzeraktionen (Christian) und das Verhalten bei gleichnamigen Orten (Christian). Es wurden zudem Regler für die OpenData (Sabrina), sowie weitere zum Ein- und Ausschalten der Transparenz der Polygone (Christian) implementiert. Des Weiteren wurden Buttons zum Löschen der auf der Karte angezeigten Elemente, zum Exportieren der Karte, zum Auswählen der gewünschten Tiles (Regina) und zum Wechsel in einen übergeordneten Stadtbezirk oder eine übergeordnete Stadt hinzugefügt (Backend Haydar, Frontend Sabrina). Zudem wurde der fertige Prototyp mit einer Erklär- bzw. Hilfeseite (Valerie) und der Suche per Rechtsklick auf der Karte (Regina) erweitert. Zuletzt wurden Vorbereitungen für die abschließende Kundenpräsentation vorgenommen.

## Ergebnis (Christian Born)

Die Interaktion mit dem Nutzer erfolgt ohne das Nachladen der gesamten Seite. Alle Daten werden mithilfe einer AJAX Schnittstelle an das Frontend gesendet, wo ausschließlich die betroffenen Elemente der Seite aktualisiert werden. So kommt die Anwendung mit einer URL aus. Einzig die Hilfeseite wurde auf eine weitere Seite ausgelagert. Abbildung 4 zeigt die Anwendung nach dem ersten Aufruf der Seite.

Im Folgenden sollen die Hauptkomponenten der Anwendung dargestellt werden. Hauptaugenmerk soll dabei auf den zur Verfügung stehenden Funktionen liegen.

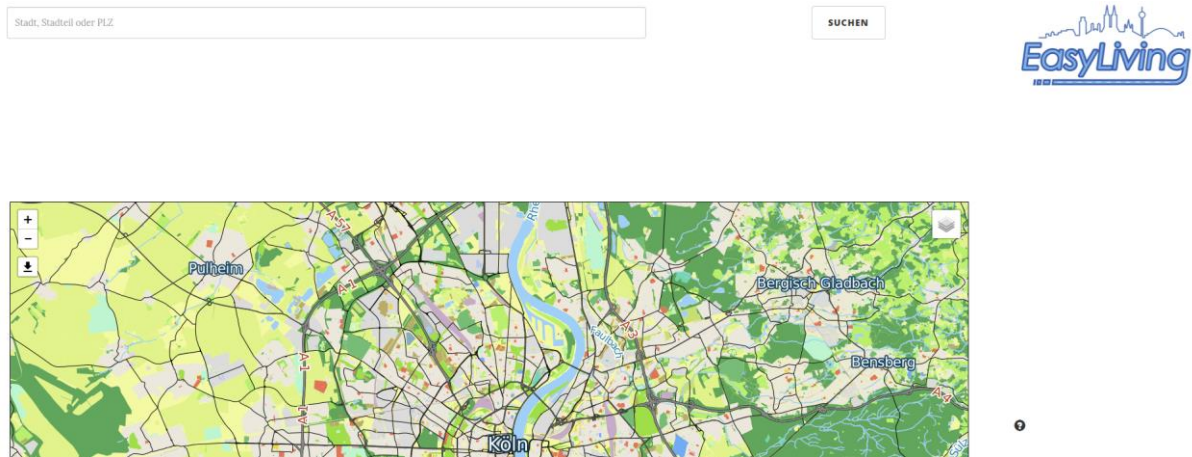


Abbildung 4: Seite beim initialen Aufruf

### Suche & Navigation

Die Suche basiert auf der Datenbasis von Open Street Maps für Nordrhein-Westfalen. Der Nutzer kann über das Suchfeld nach Städten und Gemeinden, sowie nach Stadtbezirken und Stadtteilen suchen. Hierbei muss der Suchbegriff exakt mit einem Eintrag in der Datenbank übereinstimmen, da eine Rechtschreibkorrektur oder Ähnliches nicht vorgenommen wird. Eine fehlgeschlagene Suche wird dem Nutzer durch eine Fehlermeldung mitgeteilt. Eine Suche über das Textfeld setzt den Status der Anwendung bis auf die ausgewählten Filter zurück, sodass diese Art der Suche immer den Ausgangspunkt für die weitere User Journey bildet.

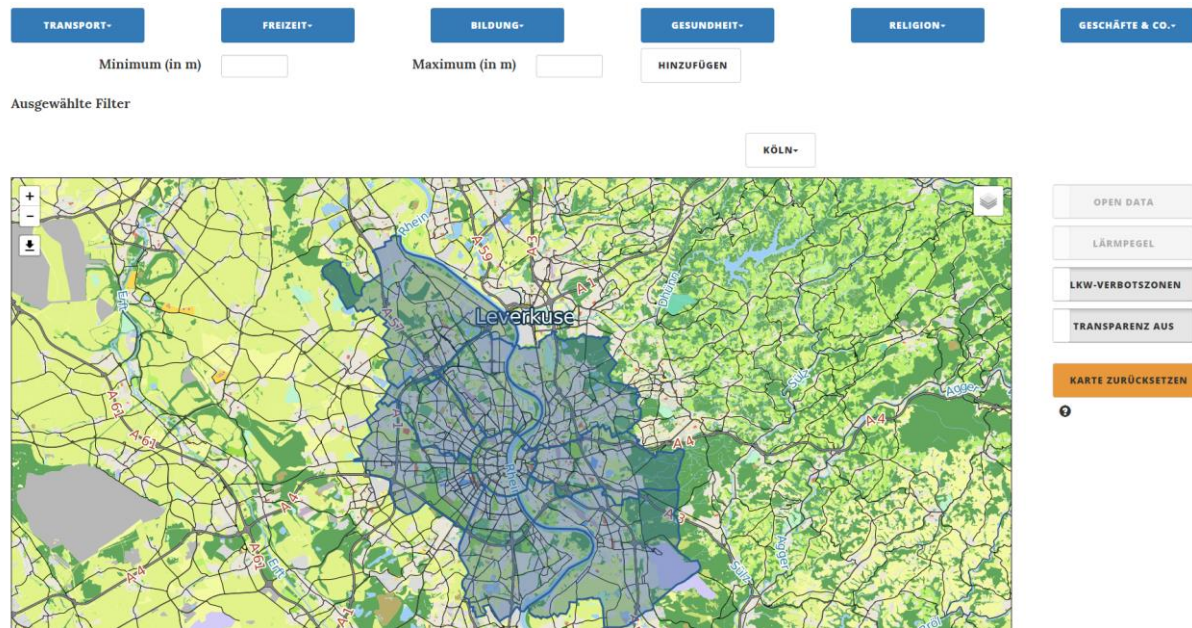


Abbildung 5: Suche nach "Köln" und Ergebnis.

War die Suche des Nutzers erfolgreich, wird ihm das gefundene Polygon auf der Karte angezeigt. Abbildung 5 zeigt eine Suche nach der Stadt Köln und das Ergebnis.

Wurden für einen Suchbegriff mehrere Einträge gefunden, werden alle gefundenen Einträge als Polygone auf der Karte markiert und in den Viewport gezoomt. Das Dropdown-Menü zeigt dem Nutzer, dass es eine ambige Treffermenge gibt und stellt ihm die Ergebnisse zur Auswahl zur Verfügung. Abbildung 6 zeigt das Dropdown-Menü und die Stadtteile des Bezirks "Köln-Innenstadt".



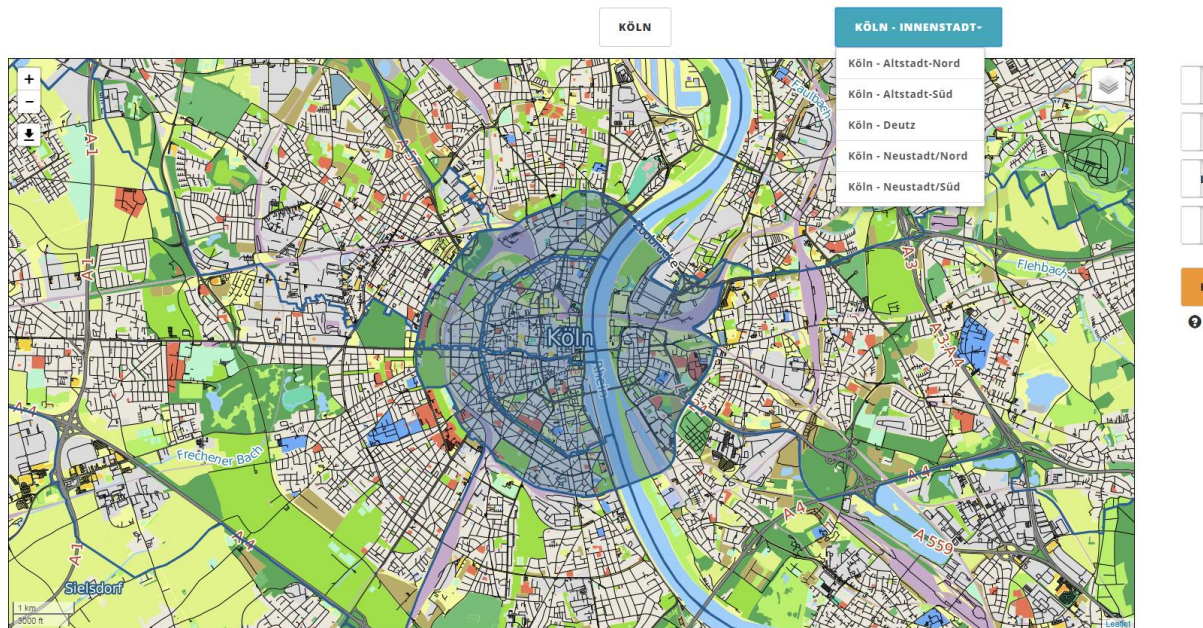


Abbildung 6: Hierarchische Navigation.

Das angesprochene Dropdown-Menü dient dem Nutzer nach einer erfolgreichen Suche über das Textfeld als weitere Navigationsmöglichkeit. Städte, Gemeinden und Stadtbezirke sind in der Regel weiter untergliedert. Ist das der Fall, stehen die Unterelemente des aktuellen Ergebnisses dem Nutzer im Dropdown-Menü zur Auswahl. Eine Auswahl eines Elements wählt das korrespondierende Polygon auf der Karte aus. Der Button links neben dem Dropdown-Menü wählt das übergeordnete Element der aktuellen Auswahl aus. Dieser Button wird ausgeblendet, wenn sich der Benutzer auf Stadtebene befindet, sodass er nicht etwa bis auf Bundeslandebene navigieren kann. So kann der Nutzer in der Hierarchie der Gliederung durch die Städte, Gemeinden, Stadtbezirken und Stadtteile rauf und runter navigieren

## Karte

Die Karte stellt die unmittelbare Visualisierung der Suchergebnisse dar. Die Darstellung kann zwischen einer Satelliten Ansicht und einer nach Orten eingefärbten Karte durch den Nutzer variiert werden. Der Bildausschnitt kann beliebig verschoben und gezoomt werden. Die Polygone des Suchergebnisses werden nach einer erfolgreichen Suche blau in der Karte eingezeichnet. Das Ergebnis-Polygon wird immer auch mit seinen nächsttieferen Unterelementen, sofern vorhanden, eingezeichnet.

Das aktuell ausgewählte Polygon erhält dabei eine farbige Füllung, während inaktive Polygone bis auf die Ränder transparent sind. Dies ist zum Beispiel der Fall, wenn der Nutzer mithilfe des Dropdown-Menüs durch die Hierarchieebenen einer Stadt navigiert.

Der Nutzer hat auch die Möglichkeit durch Interaktion mit der Karte eine Suche zu erzeugen und angezeigte Polygone auszuwählen. Klickt der Nutzer mit der rechten Maustaste auf einen Bereich der Karte, wird die dort liegende Stadt oder Gemeinde ausgewählt und als Polygon markiert. Gleichzeitig kann der Nutzer analog zum Dropdown-Menü durch die Hierarchie einer Stadt navigieren, indem er nach einer erfolgreichen Suche die Unterlemente des angezeigten Polygons mit der linken Maustaste auswählt.

Zur Erhöhung der Übersichtlichkeit kann der Nutzer die Transparenz der Polygone über einen Button neben der Karte ein- und ausschalten. Zusätzlich lassen sich die Polygone über einen Button in ihrer Gesamtheit löschen.

Falls ein Nutzer die Ergebnisse seiner Suche permanent speichern möchte, kann er über einen Button den aktuellen Kartenausschnitt als Bild herunterladen.

## Filter & Open Data

Über Filter und Open Data Datensätze kann der Nutzer mehr Informationen über das aktuell ausgewählte Polygon erhalten.

Die Filterauswahl basiert auf Landmarks aus den OSM Daten. Es stehen Filter aus den Kategorien *Transport*, *Freizeit*, *Bildung*, *Gesundheit*, *Religion* und *Geschäfte & Co.* zur Verfügung. Die Filter können über Dropdowns der entsprechenden Kategorie ausgewählt und jeweils mit einer minimalen und einer maximalen Entfernung versehen werden. Abbildung 7 zeigt ein Beispiel der Filter.



Abbildung 7: Filter-Dropdowns und Auswahl

Aus den angegebenen Entfernungen lässt sich die Schnittmenge berechnen, die den idealen Wohnort auf Basis der eingestellten Filter darstellt. Dazu werden alle ausgewählten Landmarks innerhalb des ausgewählten Polygons gesucht und die jeweils eingestellte Entfernung als Radius zur Berechnung der Schnittflächen der Radien verwendet.

Es können beliebig viele Filter und Entfernungen eingestellt werden, die zur Berechnung der Schnittfläche verwendet werden. Die Schnittfläche beinhaltet zusätzlich nur Flächen, die auch als Wohngebiete gekennzeichnet sind. So wird verhindert, dass dem Nutzer ein Wohnort ausgegeben wird, der nicht bewohnbar ist. Abbildung 8 zeigt ein Beispiel für eine Schnittflächenberechnung, die durch ein oranges Polygon visualisiert wird.

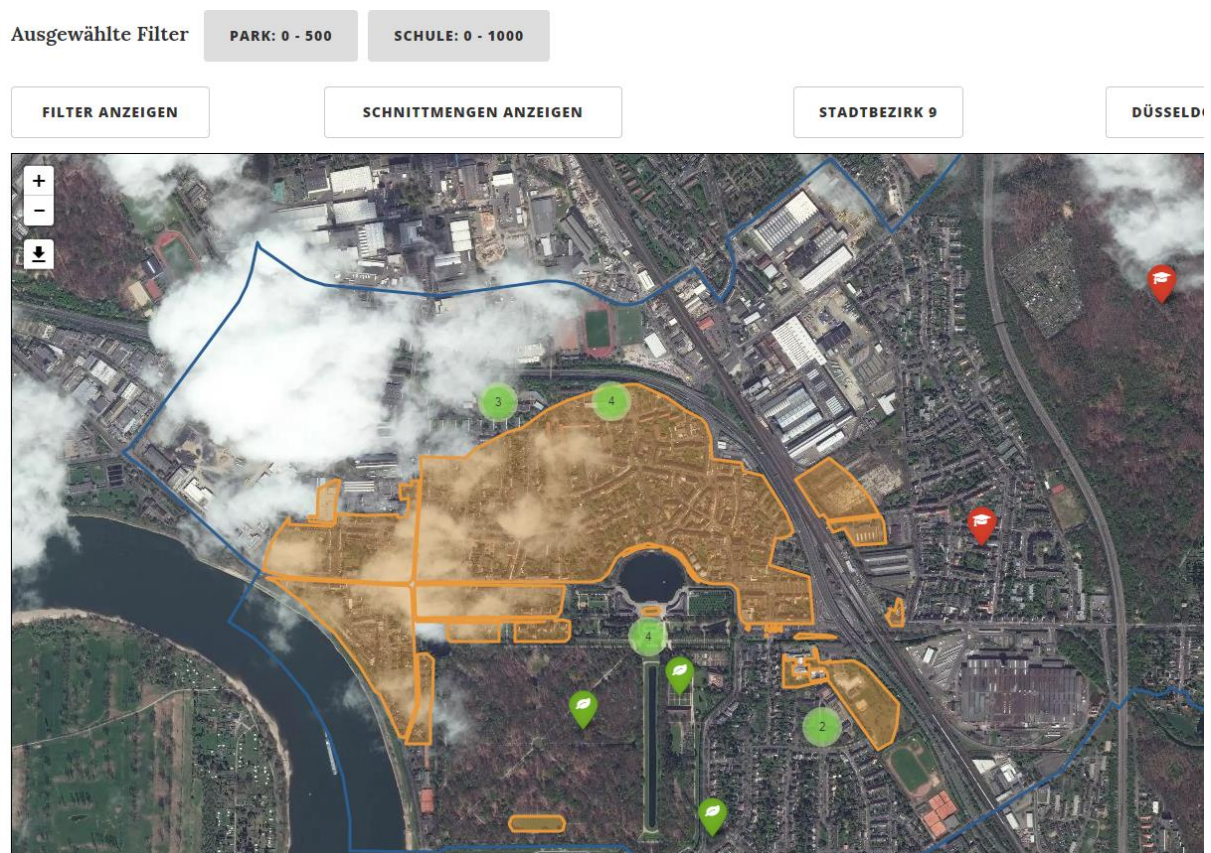


Abbildung 8: Schnittflächenberechnung und -anzeige.

Filter lassen sich auch mit unvollständigen Angaben verwenden. Lässt man das min-Feld frei, geht das System von einer minimalen Entfernung gleich 0 aus. Wird das max-Feld nicht befüllt, wird als Default eine Entfernung von 10.000 Metern verwendet. Die einzige Einschränkung bei der Nutzung der Felder ist, dass der max-Wert immer größer als der min-Wert sein muss.

Ausgewählte Filter werden dem Nutzer als Kästen jeweils mit den gewählten Entfernungen angezeigt. Durch Klick auf die Kästen lassen sich Filter wieder entfernen.



Der Nutzer hat auch die Möglichkeit, sich die ausgewählten Filter als Landmarks auf der Karte innerhalb des aktuellen Polygons anzeigen zu lassen ohne eine Schnittmengenberechnung durchzuführen.

Für die Auswahl der Filter wurde eine zusätzliche Komfortfunktion eingebaut, die dem Nutzer nur die Filter zur Verfügung stellt, die auch mindestens einmal in dem aktuellen Polygon vorkommen.

Abseits der Landmarks aus den OSM-Daten stehen dem Nutzer aus Open Data weitere Daten zur Anzeige zur Verfügung. Alle Open Data Daten sind nur für Köln verfügbar. Auf Stadtebene lassen sich Lärmpegel und LKW-Verbotzonen auf der Karte einblenden. Auf Stadtteilebene lassen sich zusätzlich zum Lärmpegel noch Daten zur Jugendarbeitslosigkeit, Arbeitslosigkeit, zum durchschnittlichen Mietpreis, Durchschnittsalter und zu den letzten Ergebnissen der Landtagswahl einschalten. Diese Daten werden bei Hover über einem Stadtteil entsprechend in einem Tooltip eingeblendet. Die Anzeige aller Open Data Daten lässt sich über Schieberegler an- und ausschalten.

Sind für das aktuelle Polygon keine Open Data Daten vorhanden, werden die entsprechenden Schieberegler ausgegraut und können nicht benutzt werden. Ein Beispiel für aktive Schieberegler sieht man in Abbildung 9.

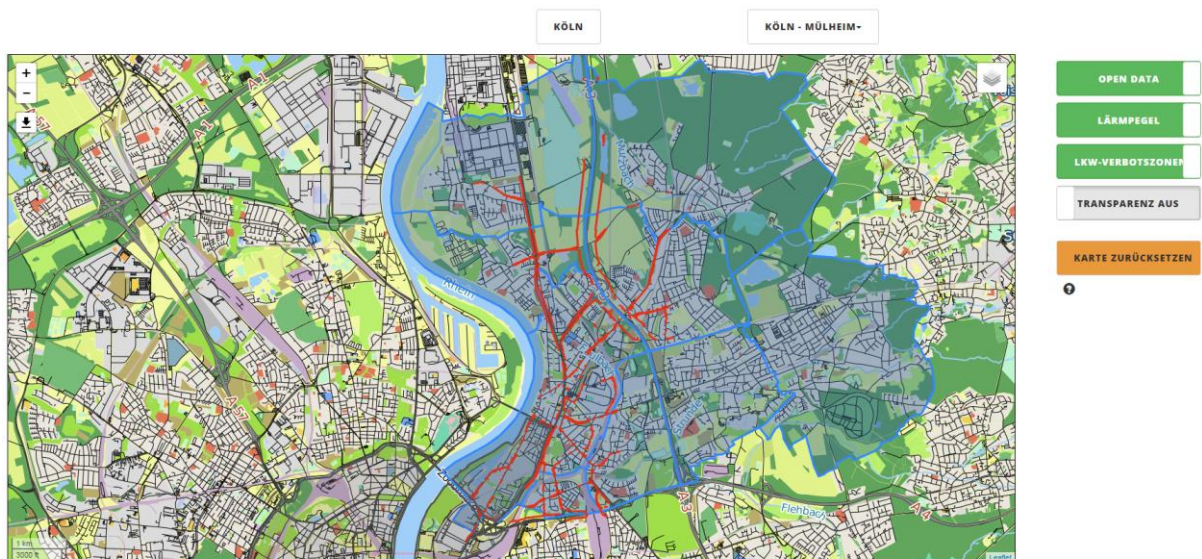


Abbildung 9: Open Data Schieberegler.

## Ausblick (Sabrina Wirkner)

### Einsatzgebiet

Durch das Alleinstellungsmerkmal der Anwendung, dass der Nutzer nach einem idealen Wohngebiet sucht anstatt die Lage von ausgewählten Gebieten zu untersuchen, bringt bereits die alleinige Anwendung von *EasyLiving* einen Nutzen. Die Verbindung mit anderen Systemen hebt die Stärken der Anwendung jedoch weiter hervor. Das Ergebnis der Anwendung ist ein markierter Bereich, der den Wohnbedürfnissen des Nutzers entspricht. Bei der Suche nach diesem Bereich handelt es sich um eine Suche nach einem Ideal - das Angebot des Immobilienmarktes wird dabei außer Acht gelassen. Aufgrund dessen ist eine Integration der Anwendung in ein bestehendes System sinnvoll. Zum Beispiel ist die Einbindung in eine Immobiliensuche denkbar. Dies würde dem Nutzer dieser Seite einen Mehrwert bieten, da er ohne großen Aufwand feststellen kann, ob Wohnungen in seinem Idealgebiet verfügbar sind oder inwiefern für ihn interessante Wohnungen seinen Bedürfnissen entsprechen. Dadurch, dass der Nutzer bereits eine Vorstellung von dem Gebiet hat, in das er ziehen möchte, wird die erste Suche auf der Immobilienseite vereinfacht und einer Überforderung des Nutzers wird entgegengewirkt. Dabei ist der Nutzer frei von den Angaben von Maklern und Verkäufern und kann selbst überprüfen, ob sich die Wohnung in der Nähe einer guten Schule befindet und Ähnliches. Die visuelle Darstellung unterstützt ferner die Nutzerfreundlichkeit und Einfachheit der Suche.

### Erweiterungsmöglichkeiten

Während der Implementierung der Anwendung fielen eine Reihe von möglichen Funktionen und Änderungen auf, welche jedoch aufgrund geringerer Priorität zeitlich noch nicht umzusetzen waren, aber eine Verbesserung für *EasyLiving* darstellen und daher hier vorgestellt werden.

Die Berechnung der Umkreise erfolgt momentan über Luftlinie, wobei die gefundenen Bereiche mit möglichen Wohngebieten geschnitten werden. Damit es sich um die echte Entfernung zu einem Punkt handelt, müssten jedoch die tatsächlichen Straßen und Wege

beachtet werden. Dies ist z.B. durch die Nutzung des Features *pgRouting* möglich, welches Entfernung auf genau diese Weise berechnet. So können sich Nutzer sicher sein, dass es nur eine bestimmte Zeit dauert, um einen Ort zu erreichen.

Wie bereits erwähnt wird bei der Ortssuche zwar eine leere Suche abgefangen und der Nutzer wird darauf hingewiesen, wenn kein Ort unter der eingegebenen Bezeichnung existiert, jedoch findet keine Autokorrektur oder -vervollständigung statt. Durch die Implementierung dieser käme es zu weniger erfolgloser Suchen und die Autovervollständigung würde einer fehlerhaften Eingabe entgegenkommen, was folglich zu einer Verbesserung der Usability führt. Des Weiteren kann die Suche insofern erweitert werden, als dass der Nutzer nicht nur nach Städten und Postleitzahlen, sondern auch nach Adressen oder Lokalitäten suchen kann. Dadurch passt sich die Anwendung zunehmend den Ortskenntnissen des Nutzers an.

*EasyLiving* arbeitet nach dem Prinzip der Suche nach einem idealen Gebiet unter angegebenen Kriterien. Zukünftig ist darüber hinaus eine Punktkreisanalyse als neue Funktion denkbar. Darunter ist einmal zu verstehen, dass der Nutzer Informationen über einen konkreten Punkt auf der Karte erhalten kann, beispielsweise welche Freizeitmöglichkeiten sich im Umkreis von einem Kilometer zu diesem befinden. Zudem macht eine umfassendere Analyse von Gebieten möglich, dass dem Nutzer bei einer erfolglosen Suche mitgeteilt wird, dass seine Suche ein Ergebnis liefern würde, wenn beispielsweise der Umkreis eines Filters erhöht wird. Dadurch bekommt der Nutzer einen Anhaltspunkt für seine nächste Suche und in welche Richtung er seine Bedürfnisse anpassen sollte. Eine solche Analyse ist allerdings sehr komplex, da sie eine Vielzahl von Möglichkeiten zur Veränderung der Suche überprüft und letztlich jene Suche mit Ergebnis vorschlägt, die der des Nutzers am ähnlichsten ist. Es muss folglich untersucht werden, wie man eine solche Punktkreisanalyse am effektivsten implementiert.

Bereits während der Ausarbeitung der Grundidee waren Suchprofile und eine Suchhistorie denkbare Funktionen für die Anwendung. Mögliche Suchprofile sind *Student*, *Familie* oder *Wohngemeinschaft*. Dabei kann der Nutzer zu Beginn angeben, dass er einer dieser Gruppen angehört und *EasyLiving* bezieht automatisch jene Kriterien in die Suche mit ein, welche im Regelfall auf diese Gruppe zutreffen, beispielsweise Nähe zu Universität und Supermarkt. Dadurch verlangt die Anwendung weniger Eigenarbeit vom Nutzer,

während dieser dennoch die Möglichkeit hat, die Kriterien auf sich anzupassen. Unter einer Suchhistorie versteht man das Speichern und Wiederaufrufen von vorigen Suchen. Da die Suchen sehr komplex werden können und es folglich Zeit kostet, diese zu wiederholen, kann es dem Nutzer helfen, wenn er auf frühere Suchen Zugriff hat, um diese zum Beispiel mit einer aktuellen Suche zu vergleichen.

Eine weitere Verbesserungsmöglichkeit bezieht sich auf den Export. Der Export der Ergebnisse beschränkt sich zur Zeit auf einen Screenshot der Karte. Er ist also rein visuell. Hilfreicher ist jedoch ein umfangreicher Export aller Parameter, also der Gesamtheit der ausgewählten Filter und ihre Umkreise, einer Legende und Angaben zum Ort und möglicher OpenData.

Allgemein ist ein Schwerpunkt zukünftiger Verbesserungen die stetige Vereinfachung der Bedienung, sodass der Nutzer bei der Suche weniger stark geleitet werden muss und die Nutzung fortwährend intuitiver wird.

Ein letzter Punkt, welcher erheblichen Einfluss auf den Informationsgehalt der Suchen hat, ist die oft nicht flächendeckende und verhältnismäßig geringe Einbindung von OpenData, welche zur Zeit lediglich für Köln implementiert sind, wobei es selbst in den Stadtteilen Kölns zu Informationslücken bezüglich der OpenData kommt. Der Grund dafür wird im nächsten Abschnitt erläutert.

## **Reflexion und Kritik**

Das größte Hindernis bei der Zusammenbringung von Open Street Maps und OpenData war die mangelhafte Verfügbarkeit der OpenData. Nur Bruchteile von für die Anwendung nützlicher Informationen sind in Form von OpenData zugänglich. Oftmals sind Daten nicht flächendeckend für NRW oder sogar für einzelne Städte verfügbar oder sie haben kaum Informationsgehalt. Ein Beispiel dafür ist der implementierte Lärmpegel. Dieser unterscheidet nur zwischen zwei verschiedenen Lärmstärken, wobei beide dieser nicht laut sind. Man bekommt also nur Zugriff auf zensierte Daten. Hinzu kommt, dass es teilweise nicht transparent ist, woher die Daten stammen oder wie aktuell diese sind. Des Weiteren werden viele für die Wohnungssuche sinnvolle Informationen wie Kriminalitätsrate zurückgehalten oder es ist fraglich, ob diese nicht zu eigenen Gunsten manipuliert

sind. Ein positives Beispiel für das Bereitstellen von OpenData ist die Stadt Köln. Obwohl auch für Köln nicht alle Daten flächendeckend sind und es zu einigen interessanten Punkten keine Informationen gibt, sind eine Reihe von OpenData einsehbar, die sich in der Regel über die Gesamtheit der Stadtteile erstrecken. Es ist wünschenswert, dass weitere Städte OpenData in dieser Form zur Verfügung stellen, um Anwendungen wie diese zu bereichern. Folglich ist es erstrebenswert, Städte zu diesem Schritt zu motivieren, indem man ihnen die Vorteile von OpenData anhand von Anwendungen wie dieser aufzeigt.

Eine zweite Problemstelle bezieht sich auf die Open Street Maps Daten. Ähnlich wie die OpenData sind viele der Daten nicht einheitlich in die Datenbank integriert. Zum Beispiel sind Daten doppelt unter verschiedenen Kategorien oder mit verschiedenen Informationen eingetragen, wobei es nicht ersichtlich ist, welcher Eintrag vollständigere Informationen bietet. Ein anderer Fall ist der Eintrag von Daten in nicht sinnvolle Kategorien. Zum Beispiel fielen mehrere Fälle auf, in denen dieselbe Schule für alle aneinandergrenzenden Gebäudeteile als Schule eingetragen ist, sodass dem Nutzer bei der Suche fälschlicherweise mitgeteilt wird, dass es in einem bestimmten Gebiet mehr Schulen gibt als tatsächlich vorhanden. Allgemein lässt sich sagen, dass die Open Street Maps-Vorgaben in einer Vielzahl der Fälle nicht eingehalten werden, indem Tags falsch gesetzt wurden oder der Eintrag falsch zugeordnet wurde. Um dieses Problem zu beheben ist eine grundlegende Überprüfung der Datenbank erforderlich, welche zeit- und arbeitsintensiv ist, jedoch die Nutzung von Open Street Maps erheblich verbessern würde.



## **Fazit (Christian Born)**

Insgesamt lässt sich festhalten, dass das Projekt sehr positiv verlaufen ist. Natürlich ist das Ergebnis ein Prototyp, der auch noch einiges an Optimierungspotenzial mit sich bringt. Auch die Datenbasis weist an einigen Stellen Verbesserungspotenziale auf, insbesondere die Open Data Daten. Doch trotz personeller Schwierigkeiten und weitestgehend fehlender Erfahrung mit einigen Technologien und der Datenbasis ist das Ergebnis äußerst zufriedenstellend. Eine Integration und Nutzung in bestehende Systeme ist mit der Optimierung des Prototypen durchaus denkbar.

## Literaturverzeichnis

Hegger, Manfred; Dammaschk, Lutz; El Khouli, Sebastian; Keller, Michael; Mahal, Nikola; Nawaz, Khalid; Petrov, Ilia; Spitzner, Kathrin: *Wohnwert-Barometer; Erfassungs- und Bewertungssystem nachhaltiger Wohnqualität*. Bauforschung in der Praxis, Band 9. Fraunhofer IRB Verlag, Stuttgart ISBN 978-3-8167-8135-6 [Buch], (2010)