

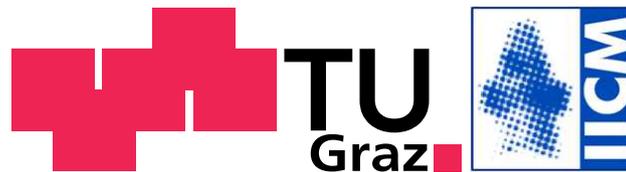
Design und Implementierung Geografischer Informationsverarbeitung in einem Social Media Portal

Bestehend aus Sport- und Freizeitaktivitäten in Verbindung mit Geocaching Elementen

Masterarbeit
an der
Technischen Universität Graz
vorgelegt von
Hannes Gritsch

Betreuer: Univ.-Doz. DI Dr.techn. Christian Gütl

Institut für Informationssysteme und Computer Medien (IICM)



Graz, September 2013

Design and Implementation of Geographical Information Processing for a Social Media Webportal

Managing Sport and Leisure Activities in Combination with Geocaching Elements

Master's Thesis

at

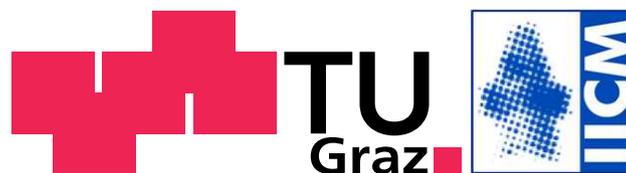
Graz University of Technology

submitted by

Hannes Gritsch

Supervisor: Univ.-Doz. DI Dr.techn. Christian Gütl

Institute for Information Systems and Computer Media (IICM)



Graz, September 2013

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Graz, _____

Date

Signature

Eidesstattliche Erklärung¹

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am _____

Datum

Unterschrift

¹Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008; Genehmigung des Senates am 1.12.2008

Kurzfassung

Das Überwachen und Auswerten von Sport- und Freizeitaktivitäten im World Wide Web oder mit Hilfe von Smartphone-Applikationen wurde in den letzten Jahren durch die Integration von GPS-Empfängern in modernen Smartphones, sowie durch billiger werdenden Empfangsgeräte immer beliebter. Durch diese Entwicklungen ist es Sportlern möglich, durch das Aufzeichnen des Streckenverlaufs eine exakte Auswertung der erbrachten Leistung zu erhalten. Alternativ dazu entwickelte sich eine Vielzahl von Geocaching-Applikationen, welche in Verbindung mit der GPS-Technik virtuelle, im Gelände durchzuführende, Schatzsuchen zum Thema haben. Ziel dieser Spiele ist es, vorgegebene Koordinaten zu finden, um dort befindliche virtuelle Gegenstände aufzusuchen und diese aufnehmen zu können.

Das Projekt, welches in dieser Masterarbeit beschrieben wird, entwickelt eine Webseite, die zum einen erlaubt, Sport- und Freizeitaktivitäten auszuwerten, zum anderen Geocaching-Funktionalität bereitstellt. Durch die Kombination dieser beiden Ansätze soll der Athlet durch die Verwendung der spielerischen Geocaching-Funktionalität zu mehr Sport motiviert werden. Mit dem Bereitstellen verschiedener Geocaching-Objekte werden dem Athleten unterschiedliche Ziele vorgegeben. Entweder muss eine gewisse Koordinate erreicht, oder eine längere Strecke bestritten werden. Zusätzlich haben die Benutzer die Möglichkeit, an virtuellen Rennen teilzunehmen. Dies motiviert die Sportler, da es einen direkten Vergleich untereinander auf einem Streckenabschnitt erlaubt.

Diese Masterarbeit befasst sich mit den Hintergründen, den Anforderungen, und der Implementierung einer Social-Community-Webseite, welche die zuvor beschriebene Funktionalität aufweist. Dafür wird in der ersten Hälfte der Arbeit auf die theoretischen Grundlagen eingegangen. Die untersuchten Themen und Gebiete zeigen, wie durch Standardisierungen Geoinformationen ausgetauscht und verwendet werden können. Dabei werden vor allem freie Dienste zur Anwendung kommen, um nicht an Lizenzbedingungen proprietärer Services gebunden zu sein. Die darin untersuchten Themen bilden die Grundlage für die Umsetzung des Projektes. Ein Teil der Implementierung befasst sich mit der Umsetzung und Darstellung der Strecken, sowie mit dem Ändern des Streckenverlaufs durch die Verwendung eines Streckeneditors. Dieser ermöglicht es den Benutzern der Webseite, alternativ zum Aufzeichnen einer Strecke mit einem GPS-Empfänger, die Route direkt auf der Webseite zu erstellen.

Mit Hilfe des Streckeneditors ist es weiteres möglich, bestehende Routen zu bearbeiten, um schnell und einfach alternative Verläufe erstellen zu können. Anschließend wird in einer kurzen Analyse der Ablauf aller wichtigen Punkte illustriert. Dies zeigt dem Leser, wie es möglich ist ein Geocaching-Objekt aufzunehmen. Zusätzlich werden die Darstellung einer virtuellen Rennstrecke und das Ranking der User gezeigt. Um neben einer Analyse aus der Benutzersicht auch ein Feedback über die Akzeptanz der Funktionalität des Webportals erhalten zu können, wurde zusätzlich ein Fragebogen ausgearbeitet. Das daraus resultierende Ergebnis zeigt, dass das Portal ein Großteils positives Feedback bekommt, mit Verbesserungspotenzial bei Design und Performance.

Zusammenfassend beschreibt dieses Projekt einen innovativen Ansatz um Sportaktivitäten und Geocaching zu verbinden. Jedoch wird ersichtlich, dass für eine gute Nutzung des Systems eine Smartphone-Applikation unumgänglich ist. Damit können Strecken direkt aufgezeichnet und den Benutzern ein sofortiges Feedback über gewonnenen Geocaching-Objekte gegeben werden.

Abstract

In the last years monitoring and evaluation of sport or leisure activities on the World Wide Web or on a smartphone application became more and more popular. One reason for this is the cheaper technique, which made GPS-receivers very popular and leads to an integration of the technique into modern smartphones. So athletes are now able to record their exercised track to get a summary of their workout performance. Beside this, the GPS-technology could be used in leisure time for geocaching applications. The applications try to build up a virtual scenario for the player. The chief aim is to find objects, which have been scattered somewhere outdoor. Therefore the coordinates of the object have to be explored with the help of a smartphone using GPS.

The project of this thesis develops a website, which allows monitoring and evaluating sport and leisure activities combined with geocaching functionality. The combination of these two aspects should motivate the athletes to do more sports and improve their own performance in an innovative way. By offering different geocaching objects, which are point or distance objects, the athlete has to reach different goals during doing his exercises. Additionally the users can participate a virtual competition on a defined racetrack. By comparing their performances among themselves the motivation of the users increases and they get better and faster.

The thesis deals with the background, the requirements and the implementation of a social community website containing the functionality described above. In the first half of this work the theoretical basics are discussed. This chapter describes how geographical information can be exchanged by standardizations. Especially free services will be monitored for this project, because proprietary services are bound by license terms. This extensive research represents the basis of the implementation process.

The implementation process could be separated into two main parts: In the first part the functionality of the display of the tracks and the redesign by using a route planner is discussed. This planner can also be used to design a track directly on the website, it is not necessary to have a GPS device. Additionally the route planner can be used to redesign existing tracks and to create different route profiles in a fast and efficient way. In the second part, the implementation of the geocaching functionality is described. It shows how the distance between two different geographical objects can be calculated.

A short usage description illustrates the most important functionality of the implemented project. It describes how a track is created using the route planner and later on how a geocaching object could be picked up. To get a better feedback of the whole portal including the route planner, a survey has been made. It consisted of general questions about the sportiness of the participators and special questions about the portal. In recap, the portal was evaluated positively, just in performance and design it could be improved.

In summary, this project describes an innovative approach to combine sports- and geocaching functionality. However, it is apparent that a smartphone application is essential for the system. In combination with a smartphone application, the athlete could get instant information of the performance immediately by recording the activity. The app could also show the geocaching objects near the user and give a fast feedback of found objects.

Danksagung

Diese Masterarbeit möchte ich meinen Eltern, Gerhard und Brigitte, widmen, da sie nicht nur mein Studium zum größten Teil finanziert haben, sondern auch ständig ein sehr großes Interesse an meiner Arbeit zeigten und mich so gut es ging in jeglicher Art und Weise unterstützt haben.

Darüber hinaus möchte ich sehr herzlich meinem, mich beinahe das ganze Studium begleitenden, Studienkollegen Jörg Sandriesser danken. Zusammen konnten wir viele schwierige Hürden im Zuge des Studiums meistern. Von ihm wurde parallel zu mir in seiner Masterarbeit die Themen Social-Media und Gamification des Webportals entwickelt. Wir danken unserem Betreuer dafür, dass wir ein umfangreiches und sehr interessantes Projekt gemeinsam entwickeln durften.

Folglich gilt besonderer Dank Herrn Dipl.-Ing. Dr.techn. Christian Gütl, der mir die Möglichkeit gegeben hat, diese Arbeit zu erstellen und mich während der Masterarbeit betreut und umfangreich unterstützt hat. Darüber hinaus möchte ich noch Herrn Pascal Nies, Msc. von der Universität Heidelberg danken, der mir erlaubte, einen vom ihm entwickelten Routenplaner in der Umsetzung dieses Projektes verwenden zu dürfen. Dankend angenommen wurde auch die Unterstützung von Herrn Lauri Frank und Herrn Paul Magiatis, durch deren Input neue Ideen und Funktionen für dieses Projekt gefunden werden konnten.

Hannes Gritsch
Graz, Juli 2013

Inhaltsverzeichnis

| | | |
|----------|--|-----------|
| 1 | Einleitung | 1 |
| 1.1 | Motivation und Ausgangssituation | 1 |
| 1.2 | Ziele | 2 |
| 1.3 | Struktur der Arbeit | 2 |
| 2 | Einführung in die Geoinformationssysteme | 4 |
| 2.1 | Geschichtliche Entwicklung von Geoinformationssystemen | 4 |
| 2.2 | Standardisierungsbemühungen für Geoinformationen | 5 |
| 2.3 | Dienste für Geoinformationen | 7 |
| 2.3.1 | Web Map Service | 7 |
| 2.3.2 | Web Feature Service | 8 |
| 2.3.3 | Open Location Services | 8 |
| 2.4 | Modellierung von geographischen Daten | 10 |
| 2.4.1 | Vektordaten | 10 |
| 2.4.2 | Rasterdaten | 11 |
| 2.4.3 | Hybride | 13 |
| 2.5 | WebGIS | 13 |
| 2.6 | Geosemantik | 14 |
| 2.7 | Zusammenfassung | 16 |
| 3 | Ausgewählte Themen | 17 |
| 3.1 | Globale Navigationssatellitensysteme | 17 |
| 3.1.1 | Entwicklung von Satellitennavigation | 17 |
| 3.1.2 | Funktionsweise der Satellitenortung | 18 |
| 3.1.3 | Einsatzbereiche von Satellitennavigation | 19 |
| 3.2 | Proprietäre und freie Geodaten | 20 |
| 3.2.1 | Google Maps | 21 |
| 3.2.2 | OpenStreetMaps | 22 |
| 3.2.3 | Freie Geocoding Dienste | 25 |
| 3.2.4 | Freie Routenplaner auf Basis der OpenStreetMap Karten | 28 |
| 3.2.5 | Freie Höhendatendienste | 29 |
| 3.3 | Zusammenfassung | 30 |
| 4 | Aktuelle Anwendungen | 31 |
| 4.1 | Webportale und Apps zur Analyse von Sport- und Freizeitaktivitäten | 32 |
| 4.1.1 | <i>Runtastic</i> | 32 |
| 4.1.2 | <i>Endomondo</i> | 34 |
| 4.1.3 | <i>Strava</i> | 36 |
| 4.2 | Ausgewählte Tools mit Geocaching Funktionalität | 36 |
| 4.3 | Zusammenfassung | 38 |

| | | |
|----------|--|-----------|
| 5 | Anforderungen und Design | 39 |
| 5.1 | Analyse allgemeiner Anforderungen | 39 |
| 5.2 | Funktionale Anforderungen | 40 |
| 5.2.1 | Präsentation von Routen und Geocaching-Objekten | 40 |
| 5.2.2 | Berechnung der Routendaten | 41 |
| 5.2.3 | Routenplaner und Höhenmeter | 41 |
| 5.2.4 | Wiederverwendung und Änderung von Routen | 41 |
| 5.2.5 | Virtuelle Rennstrecken | 42 |
| 5.2.6 | Geocaching von Punkten und Strecken | 42 |
| 5.2.7 | Zusätzliche Hilfen | 42 |
| 5.3 | Nichtfunktionale Anforderungen | 42 |
| 5.3.1 | Aussehen und Usability | 43 |
| 5.3.2 | Leistung und Effizienz | 43 |
| 5.3.3 | Erweiterbarkeit | 43 |
| 5.3.4 | Flexibilität | 43 |
| 5.3.5 | Korrektheit | 44 |
| 5.3.6 | Portierbarkeit und Übertragbarkeit | 44 |
| 5.3.7 | Sicherheitsanforderungen | 44 |
| 5.4 | Konzeptionelle Architektur | 44 |
| 5.4.1 | Komponenten im Webportal | 44 |
| 5.4.2 | Erstellen einer Aktivität | 46 |
| 5.5 | Zusammenfassung | 47 |
| 6 | Tools und Framework | 48 |
| 6.1 | Entscheidungsfindung | 48 |
| 6.2 | Drupal 7 | 49 |
| 6.3 | OpenLayers | 50 |
| 6.4 | Highcharts for JS | 52 |
| 6.5 | Zusammenfassung | 52 |
| 7 | Implementierung | 54 |
| 7.1 | Grundlagen zu Drupal 7 | 54 |
| 7.1.1 | Grundlegende Terminologie | 54 |
| 7.1.2 | Architektur und Funktionalität | 56 |
| 7.2 | Die OpenLayers Grundlagen | 56 |
| 7.2.1 | Map-Objekt | 56 |
| 7.2.2 | Layer-Objekt | 57 |
| 7.2.3 | Control-Objekt | 57 |
| 7.3 | Architektur und Aufbau des neuen Webportals | 59 |
| 7.4 | Verarbeitung und Präsentation der Streckendaten | 60 |
| 7.4.1 | Erstellen einer Route durch Einbindung eines Routenplaners | 60 |
| 7.4.2 | Präsentation von Daten einer GPX-Datei | 66 |
| 7.4.3 | Präsentation einer bestehenden Strecke aus dem Drupal-System | 68 |
| 7.4.4 | Darstellung des Höhenprofils | 69 |
| 7.4.5 | Berechnung von Distanz, Höhe und Geschwindigkeit | 70 |
| 7.4.6 | Speicherung der Daten im Drupal-System | 71 |
| 7.5 | Umsetzung des Geocaching | 72 |
| 7.5.1 | Erstellen eines Geocaching-Objekts | 72 |

Inhaltsverzeichnis

| | | |
|---------------------------------------|---|------------|
| 7.5.2 | Ablauf des Geocachings | 72 |
| 7.5.3 | Berechnung der Entfernung von geografischen Objekten zueinander . . . | 73 |
| 7.5.4 | Berechnung von Zeitintervallen auf Streckensegmenten | 75 |
| 7.6 | Verwendung von OpenLayers in Drupal 7 | 76 |
| 7.7 | Zusammenfassung | 78 |
| 8 | Analyse aus Benutzersicht | 80 |
| 8.1 | Die Übersichtskarte | 80 |
| 8.2 | Anlegen einer Aktivität | 82 |
| 8.2.1 | Erstellen einer Route | 82 |
| 8.2.2 | Ändern einer bestehenden Strecke | 84 |
| 8.3 | Darstellung einer Aktivität | 85 |
| 8.4 | Ablauf von Geocaching | 85 |
| 8.5 | Evaluierung des Webportals | 87 |
| 8.5.1 | Aufbau des Fragebogens | 87 |
| 8.5.2 | Stakeholder Gruppe | 88 |
| 8.5.3 | Auswertung | 88 |
| 8.6 | Zusammenfassung | 92 |
| 9 | Lessons Learned | 93 |
| 9.1 | Theoretischer Teil | 93 |
| 9.2 | Praktische Umsetzung | 93 |
| 9.3 | Interpretation der Benutzerbefragung | 94 |
| 10 | Zusammenfassung und Ausblick | 95 |
| Anhang A | | 97 |
| Literatur | | 98 |
| Abbildungsverzeichnis | | 105 |
| Tabellenverzeichnis | | 106 |
| Listings | | 107 |
| Anhang B | | 108 |
| Inhalt der beiliegenden CD-ROM | | 109 |

1 Einleitung

Im Zuge dieser Arbeit wird gezeigt, wie es mit vorhandenen und frei zur Verfügung stehenden Frameworks und Diensten möglich ist, Verarbeitungs- und Präsentationssaufgaben von Geodaten in einer Web 2.0 Social Media Umgebung zu realisieren. Diese Social-Community wird speziell für ambitionierte Hobby- und Freizeitsportler entwickelt, die eine neue Art der Motivation suchen, öfter und regelmäßiger Sport zu treiben. Diese Arbeit besteht aus einem theoretischen und einem praktischen Abschnitt. Ersterer befasst sich mit den Grundlagen von Geoinformationssystemen, deren Standardisierungsbemühungen und Einsatzgebieten. Durch die Analyse aktueller Anwendungen werden Punkte herausgefiltert, welche als Basis für das Projekt gesehen werden können. Die Umsetzung dieser Punkte und des gesamten Projektes wird im praktischen Teil beschrieben. Die Implementierung der Social-Community, in welche dieses Projekt eingebettet wurde, kann bei Sandriesser (2013) eingesehen werden und ist nicht Teil dieser Arbeit.

1.1 Motivation und Ausgangssituation

Laut Jansen und Adams (2010, S. 14) erfahren Geodaten und geografisches Kartenmaterial im World Wide Web immer größer werdender Beliebtheit. Zum einen liegt das daran, dass die angebotenen Karten besser auf die Bedürfnisse der User zugeschnitten werden können. Das bedeutet, es können auf ihnen die verschiedensten Arten von Informationen dargestellt werden. Zum anderen sind diese Karten, wie z.B. das OpenSource-Projekt OpenStreetMap¹, durch aktive Mitarbeit der User immer auf dem neuesten Stand. Da mit Hilfe von Smartphones mit Internetzugang jede Information immer und überall zur Verfügung steht, können auch digitale Karten jederzeit abgerufen werden. Durch Zugang zum digitalen Kartenmaterial können sofort Informationen zum aktuellen Standort abgefragt werden, sowie welche POI (»Points of Interests«) sich in der Nähe befinden. Die ständig weiter voranschreitende Entwicklung von Location Based Services (LBS), wie die erwähnten POI's, Routenplaner oder Geotagging Tools, ist nach Yoshida, Song und Raghavan (2010) den immer billiger werdenden GPS (Global Positioning System) Geräten, wie z.B. Smartphones, zu verdanken.

Nachdem im Zuge dieser Arbeit ein Sport- und Freizeitportal entwickelt wird, zeigt die Analyse bereits vorhandener Sport Community Webseiten wie Runtastic² oder Endomondo³ eine gewisse Ähnlichkeit bei der Arbeit mit Geodaten. Bei dem neuen Projekt wird versucht, die angebotenen Funktionen zu verbessern, bzw. neue Funktionen zu integrieren, die das Bearbeiten von Geodaten dem User noch einfacher ermöglichen sollen. Innerhalb dieses Projektes wird ein Webportal entwickelt, auf dem durch eine Community Outdoor-Sport-Aktivitäten

¹<http://openstreetmap.org> – zuletzt abgerufen am 03.04.2013.

²<http://www.runtastic.com> – zuletzt abgerufen am 26.04.2013.

³<http://www.endomondo.org> – zuletzt abgerufen am 09.04.2013.

1 Einleitung

angelegt und geteilt werden können. Ein besonderer Motivator dieser Seite ist es, dass durch diese Sportaktivitäten Geocaching-Objekte eingesammelt werden können. Diese Geocaching-Objekte können von der Community ausgestreut werden, um so dem einzelnen User ein abwechslungsreiches Training zu ermöglichen.

1.2 Ziele

Ziel dieser Masterarbeit soll es sein, dem Leser die Grundlagen der Geoinformationssysteme, die Entscheidungsfindung und die daraus resultierende Entwicklung eines Webportals für Sport- und Freizeitaktivitäten zu präsentieren. Der Implementierungsprozess zeigt zunächst die verwendete Drei-Schichte-Architektur als Basis für das Webportal. Des Weiteren werden alle Schritte dargestellt, welche unternommen werden müssen, um Streckendaten erstellen und bearbeiten zu können. Der zweite große Teil der Implementierung beschreibt die Verwendung von Geocaching und deren verschiedenen Funktionen.

1.3 Struktur der Arbeit

Diese Masterarbeit beschreibt die Techniken der eingesetzten Tools, sowie die Umsetzung der Funktionalität. Die ersten Kapitel beschäftigen sich mit dem theoretischen Hintergrund der Arbeit. Hier werden die Grundlagen von Geoinformationssystemen beschrieben, sowie ausgewählte Themen wie die Satellitennavigation oder Geodaten im World Wide Web. Im zweite Teil wird die Implementierung näher erläutert und ein Ausblick auf zukünftige Arbeiten gegeben.

Das Kapitel 2 stellt das Grundlagenkapitel dar und gibt eine Einführung in die Geoinformationssysteme. Darin wird auf die geschichtliche Entwicklung dieser Systeme und auf die daraus resultierenden Standardisierungsbemühungen eingegangen. Der Abschnitt zeigt, auf welche Arten geographische Daten dargestellt und ausgetauscht werden können.

In Kapitel 3 werden ausgewählte Themen in Bezug auf Geoinformationen behandelt. Darin wird u. a. die Verwendung der Satellitennavigationstechnik beschrieben. Des Weiteren wird auf den Unterschied zwischen freien und proprietären Geodaten eingegangen. Für jede dieser Kategorien wurden Beispiele erarbeitet und deren Verwendung gezeigt.

Aktuelle Anwendungen, welche ähnliche Funktionalitäten haben, wie das im Zuge dieser Masterarbeit entwickelte Projekt, werden in Kapitel 4 erörtert und untersucht. Hier wird auf Webportale eingegangen, die zur Analyse von Sport- und Freizeitaktivitäten herangezogen werden können. Zusätzlich werden Geocaching-Applikationen und deren unterschiedliche Einsatzmöglichkeiten betrachtet.

Kapitel 5 geht auf die Anforderungen und das Design für das neu entwickelte Webportal ein. Dabei werden die im Kapitel 4 betrachteten Anwendungen auf ihre Vor- und Nachteile analysiert. Daraus werden dann funktionale und nichtfunktionale Anforderungen abgeleitet sowie eine erste konzeptionelle Architektur erstellt.

1 Einleitung

Bevor die Implementierung beschrieben werden kann, zeigt Kapitel 6 die eingesetzten Tools und Frameworks. Darin wird zunächst die Entscheidungsfindung diskutiert, aus der das Content-Management-System Drupal als Basis für das Webportal hervorging. Für die Implementierung der Präsentation und Bearbeitung von Geodaten wurde OpenLayers verwendet.

Alle in der Implementierung vorgenommenen Schritte werden in Kapitel 7 beschrieben. Basierend auf der Architektur des Systems wird zunächst auf die Grundlagen von Drupal und OpenLayers eingegangen. Des Weiteren werden die notwendigen Schritte gezeigt, wie Routendaten auf Karten in Verbindung mit einem Höhenprofil Diagramm umgesetzt werden können. Um für das Geocaching wichtige Distanzen zwischen Objekten berechnen zu können, wurde ein Exkurs in Trigonometrie unternommen und der Satz des Heron beschrieben. Um alle vorhandenen Geodaten auf dem Portal darstellen zu können, wurde die Arbeit von OpenLayers innerhalb von Drupal beschrieben.

Kapitel 8 gibt eine Analyse des Webportals aus der Benutzersicht, die den Ablauf der wichtigsten Komponenten illustrieren. Darin wird auf die Möglichkeiten und Funktionen des Routenplaners eingegangen. Des Weiteren wird der Ablauf von Geocaching auf dem Webportal dargestellt. Zusätzlich wird in diesem Abschnitt auf den Fragebogen eingegangen, welcher zur Evaluierung des Portals ausgearbeitet wurde.

Das Kapitel 9 betrachtet die Erfahrungen und Erkenntnisse, die im theoretischen sowie im praktischen Teil gemacht wurden. Hier wird von den auftretenden Schwierigkeiten berichtet, welche im Zuge der Implementierung gelöst werden mussten. Kapitel 10 gibt eine Zusammenfassung sowie einen Ausblick auf Ideen für zukünftige Arbeiten.

2 Einführung in die Geoinformationssysteme

Am Beginn dieser Arbeit sollen die grundlegenden Entwicklungen und Funktionsweisen von Geoinformationssystemen (GIS) näher erläutert werden. Das folgende Kapitel beleuchtet die Entwicklung von GIS, deren Einsatzbereiche, Standardisierungsbemühungen von Geodaten und Modellierungsmöglichkeiten von Geodaten. Des Weiteren werden Services wie das Web Map Service und das Web Feature Service erläutert, sowie die Architektur des WebGIS. Abschließend wird die Geosemantik angesprochen, ein Forschungsbereich, welcher versucht Geodaten mit semantischen Informationen besser und genauer zu definieren.

Mit der Abkürzung GIS werden in der Literatur Geoinformationssysteme bzw. geographische Informationssysteme beschrieben. Ein Begriff, der laut Jansen und Adams (2010, S. 26) für jedes Programm steht, welches geographische raumbezogene Daten verarbeitet. Bei Minar (2010, S. 14) wird GIS beschrieben als „[...] ein rechnergestütztes System, das aus Hardware, Software, Daten und Anwendungen besteht. Mit ihm können raumbezogene Daten digital erfasst und redigiert, gespeichert und reorganisiert, modelliert und analysiert sowie alphanumerisch und grafisch präsentiert werden [...]“. Die Anwendungsgebiete von GIS sind sehr weitreichend und dessen Programme können in den verschiedensten Bereichen der Gesellschaft gefunden werden. Die ersten GIS Anwendungen haben sich in den Jahren von traditionellen Bereichen wie Landvermessungen und Kartographien hin zu massentauglichen Anwendungen entwickelt (Jia & Han, 2010). Geographische Informationssysteme bestehen nicht nur aus Karten und Positionen auf den Karten, viel mehr werden dynamische sozial-ökonomische Daten bearbeitet und präsentiert.

2.1 Geschichtliche Entwicklung von Geoinformationssystemen

In den 1980er Jahren wurden Geoinformationssysteme hauptsächlich für Aufgabenstellungen von Regierungen oder für militärische Zwecke eingesetzt. Andere Märkte, wie das Transport- und Ingenieurwesen, erkannten jedoch das Potential und begannen GIS Technologien zu erforschen und zu entwickeln. Die Herstellung und Wartung von guten Geodaten, wie z. B. Luftbildaufnahmen, war und ist sehr kostspielig und aufwendig, was dazu führte, dass Kunden von GIS Daten bei verschiedenen Firmen die gewünschten Informationen beantragen mussten, da oft nicht bei einer Firma allein alle gewünschten Daten vorhanden waren. Dies führte zu Problemen, da es keine einheitlichen Datenformate und Richtlinien gab, und jeder Hersteller von GIS Daten sein eigenes proprietäres Format verwendete. Der Datenaustausch war daher mit sehr hohem Aufwand verbunden, da die Daten in das richtige Format konvertiert werden mussten, was wiederum die Gefahr von Informationsverlust mit sich führte und mit hohen Kosten verbunden war. Diese Tatsachen haben den Bedarf geschaffen, auch im GIS-Bereich Standards zu definieren und zu verwenden (OGC, 2013).

Viele Länder und Städte bieten eigene GIS Informationen zum Download an, wo wichtige Punkte verzeichnet sind, wie Spielplätze oder öffentliche Toilettenlagen. Als Beispiel soll hier ViennaGIS¹ der Stadt Wien genannt werden. Unter anderem werden hier Daten wie Bezirksgrenzen, Bebauungspläne oder auch Standorte für Hundezonen und Hundeausläufe zum Download angeboten. Neuere GIS Systeme haben sich aus der Integration von GPS Empfängern in Mobiltelefone bzw. dem immer günstiger werdenden GPS Empfängern entwickelt. Die daraus entstandenen Anwendungen wären das Geotagging, wo an Dokumenten Geoinformationen hinzugefügt werden (z. B. ein Foto mit den Koordinaten wo dieses aufgenommen wurde), Geocaching, eine virtuelle Schnitzeljagd oder Schatzsuche, oder auch Routenplaner, die nicht nur im Kraftfahrzeugbereich Verwendung finden, sondern auch für Fahrräder optimiert werden können. Im Verlauf dieser Arbeit wird auf diese Begriffe noch näher eingegangen.

2.2 Standardisierungsbemühungen für Geoinformationen

Bei IT Systemen mit serviceorientierter Architektur wird versucht, durch Schnittstellen dem Benutzer exakt auf die Anfrage abgestimmte Daten bereit zu stellen. Wie diese Daten generiert und gespeichert, von welchen Datenbanken sie abgefragt oder wie sie berechnet werden, soll für den Benutzer nicht von Interesse sein. Eine Standardisierung und Definition der Schnittstellen ist daher sehr wichtig. Standardisierte Schnittstellen ermöglichen eine vollständige Kompatibilität zwischen Anwendungen. Diese erhöht die Flexibilität durch eine mögliche Wiederverwendung bestehender Services, was wiederum die Kosten von neuer Softwareentwicklung senkt (Jansen & Adams, 2010, S. 28).

Bei der Standardisierung von Geoinformationssystemen wird in der Literatur an erster Stelle das Open Geospatial Consortium (OGC²) genannt, welches die führende Organisation in der Entwicklung für Standards von Geodaten ist. Eng in Verbindung mit dem OGC arbeitet die ISO/TC211³ Normierungsbehörde. In Folge soll auf diese, sowie auf die CEN TC287⁴, welche innerhalb der Europäischen Union für europaweite Standards zuständig ist, sowie auf das World Wide Web Consortium (W3C⁵) eingegangen werden.

Open Geospatial Consortium

Das Open Geospatial Consortium (OGC) ist eine gemeinnützige und auf freiwilliger Basis arbeitende Organisation, welche Spezifikationen zur Standardisierung von Schnittstellen und Datenformaten raumbezogener Daten entwickelt (Shen, Duan, Ren & Yang, 2010). Diese Standards reichen von der abstrakten Beschreibung des Aufbaus der Komponenten und deren Funktionen bis hin zu detaillierten Implementationen dieser Dienste. Vorgeschrieben wird durch einen Standard die Einhaltung von Schnittstellen und Verhalten, jedoch nicht die konkrete Umsetzung der Software. Im OGC Prozess⁶ werden zunächst Kompatibilitätsprobleme

¹<http://www.wien.gv.at/viennagis/> – zuletzt abgerufen am 03.04.2013.

²<http://www.opengeospatial.org> – zuletzt abgerufen am 03.04.2013.

³<http://www.isotc211.org/> – zuletzt abgerufen am 03.04.2013.

⁴<http://www.centc287.eu/> – zuletzt abgerufen am 03.04.2013.

⁵<http://www.w3.org/> – zuletzt abgerufen am 03.04.2013.

⁶<http://www.opengeospatial.org/ogc/process> – zuletzt abgerufen am 03.04.2013.

2 Einführung in die Geoinformationssysteme

analysiert, welche von den OGC Mitgliedern eingebracht werden. Als Resultat dieser Analyse werden die benötigten Standards und Services definiert. Dieser Vorgang ist unumgänglich, da die Server-Client Architektur eine Standardisierung der Anfragen und Antworten voraussetzt (Lu, 2006). Zu den am häufigsten verwendeten Standardisierungen gehören die XML-basierende Geography Markup Language (GML), die Geometrieformate Well-Known Text (WKT) und Well-Known Binaries (WKB), sowie Dienstschnittstellen oder Open Web Services (OWS) wie Web Map Service (WMS), Web Feature Service (WFS) oder OpenLS (Jansen & Adams, 2010, S. 29).

Das OGC besteht aus 477 Mitgliedern (Stand: 26.03.2013)⁷, welche aus Regierungsbereichen, Privatindustrien oder von Universitäten kommen. Mitglied wird man durch einen Mitgliedsbeitrag. Die Tatsache, dass beim OGC konkurrierende Unternehmen sich auf einheitliche Standards einigen, zeigt die Wichtigkeit dieser Organisation auf (Jansen & Adams, 2010, S. 29).

ISO/TC211

Die internationale Organisation für Normung (ISO) ist im Technischen Komitee Nr.211 für die Normierung von geographischen Informationen und Geomatik zuständig. Die Hauptaufgabe der ISO Normierung liegt bei digitalen Geoinformationen und deren abstrakte Spezifikationen, wobei das OGC Definitionen von Schnittstellen und Implementierungsspezifikationen von Geoinformationssystemen erarbeitet. Im Gegensatz zum OGC, wo ein Standard durch eine Konsensentscheidung der Mitglieder definiert wird, wird bei der ISO durch einen mehrjährigen Normungsprozess des Technischen Komitees der Standard gebildet. Da von beiden Organisationen durch diesen getrennten Vorgang unterschiedliche Standards entstehen würden, wurde 1998 eine Übereinkunft über eine enge Zusammenarbeit getroffen. Dies führt dazu, dass Spezifikationen, welche von der OGC definiert werden, formell durch den ISO Abstimmungsprozess gehen um auch als ISO Standard definiert werden zu können (Klopper, 2005).

CEN TC287

Das Europäische Komitee für Normung (CEN, französisch Comité Européen de Normalisation) entwickelt und definiert Standards für den europäischen Raum. Auf freiwilliger Basis werden Normen von den 33 CEN-Mitgliedern definiert (Stand: 26.03.2013). Die Aufgabe des Technischen Komitees 287 liegt bei der Standardisierung von Geoinformationen für Europa. Diese Spezifikationen werden in enger Zusammenarbeit mit der ISO/TC211 entwickelt (Klopper, 2005).

World Wide Web Consortium

Das World Wide Web Consortium (W₃C) ist ein internationales Komitee, welches Web Standards entwickelt. Mit dem OGC unterhält das W₃C einen intensiven Informationsaustausch um Themen und Definitionen beidseitiger Interessen zu diskutieren. Diese Zusammenarbeit führte zu Standards, definiert vom W₃C wie z. B.: SVG, XML, XSLT, SOAP, WSDL, RDF und

⁷<http://www.opengeospatial.org/ogc/members> – zuletzt abgerufen am 03.04.2013.

2 Einführung in die Geoinformationssysteme

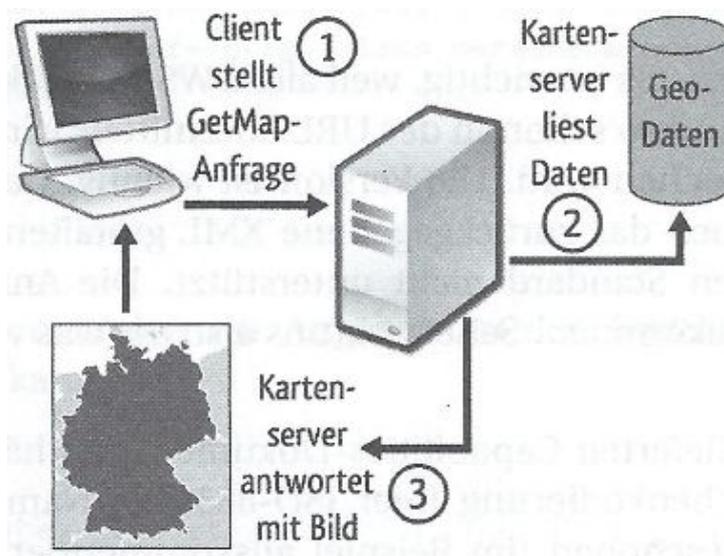


Abbildung 2.1: Schematische Darstellung der Funktionsweise eines WMS. Der Client stellt eine Anfrage an den Server, welcher die Daten aus der Datenbank aufbereitet und an den Client zurück sendet (Jansen & Adams, 2010, S. 41).

OWL. Diese Standards werden u. a. vom OGC zur Standardisierung von Geoinformationen eingesetzt, wie z. B. die Formate RDF und OWL, welche Einsatz in der Geosemantik finden.

2.3 Dienste für Geoinformationen

Laut den Autoren Shen et al. (2010) handelt es sich beim Web Map Service (WMS⁸) und dem Web Feature Service (WFS⁹) um zwei Basisservices beim Arbeiten mit GIS. Sie sind dafür verantwortlich, dass Karten bzw. Geodaten dargestellt werden können. Zusätzlich zu diesen beiden Services soll das Open Location Services (OpenLS), welches für standortbezogene Dienste zuständig ist, näher beschrieben werden (Klopfer, 2005).

2.3.1 Web Map Service

Über das Web Map Service können Kartenbilder angefordert werden. Diese Karten werden vom Server durch Daten aus einer GIS Datenbank generiert. Die meisten GIS Anwendungen können von verschiedenen Servern Geodaten einbinden (Guo, Wu, Zhang & Xiang, 2012).

Der Ablauf einer WMS Anfrage kann, wie in Abbildung 2.1 dargestellt, in drei Abschnitte unterteilt werden:

1. Der Client stellt eine Anfrage und übergibt Parameter über eine definierte URL an einen WMS Server.

⁸<http://www.opengeospatial.org/standards/wms> – zuletzt abgerufen am 03.04.2013.

⁹<http://www.opengeospatial.org/standards/wfs> – zuletzt abgerufen am 03.04.2013.

2 Einführung in die Geoinformationssysteme

2. Der Server übernimmt die Anfrage und greift auf Geodaten zu. Dieser Vorgang kann von Server zu Server unterschiedlich sein, da jede WMS-Server-Software Unterschiede aufweisen kann.
3. Die angefragten Daten werden zum Client zurückgeschickt. Meist ist es ein Bild im Format PNG, JPEG oder GIF (Jansen & Adams, 2010, S. 40).

Die wichtigsten Parameter, welche ebenfalls durch das OGC definiert wurden, sind (de La Beaujardière, 2012, S. 1):

- *GetCapabilities*: Liefert Metadaten und Informationen über den WMS-Server im XML Format zurück.
- *GetMap*: Liefert ein Kartenbild mit genau definierten Geodaten und Dimensionen zurück.
- *GetFeatureInfo*: Ein optionaler Parameter, welcher zusätzliche WMS-Server spezifische Daten bereitstellen kann.

2.3.2 Web Feature Service

Das Web Feature Service ist dem WMS sehr ähnlich. Der Unterschied besteht darin, dass hier kein Kartenbild vom Server zurückgegeben wird, sondern Geodaten im GML-Format¹⁰, ein speziell für Geodaten entwickeltes XML-Schema. Daher können durch das WFS verschiedene Geoinformationen präsentiert werden, wie z. B. Landesgrenzen, Flüsse oder Städtenamen. Das Format dieser Daten wird in Vektorform bereitgestellt, wobei beim WMS Rasterdaten verwendet werden (siehe Abbildung 2.2). Zusätzlich ist es mit WFS möglich, nicht nur Geodaten anzufordern, sondern auch Geodaten auf den Server zurückzuschreiben. Clients können selbst veränderte Objekte wieder auf den WFS Server schreiben. Dies setzt spezielle Einstellungen am Server voraus, wie z. B. das Verhindern, dass mehrere Personen gleichzeitig die selben Objekte bearbeiten (Jansen & Adams, 2010, S. 55).

Bei Mitchell (2008, S. 254) werden die unterschiedlichen Verwendungszwecke des WFS im Vergleich zum WMS beschrieben. Das Web Feature Service bietet die Möglichkeit, nach speziellen Objekten zu suchen wie z. B. den Namen einer Stadt. Es können aber auch Suchergebnisse, wie Stadtgrenzen etc. markiert werden.

2.3.3 Open Location Services

Das *Open Location Services* (OpenLS) wurde im Jahr 2000 von der OGC entwickelt, um Schnittstellen und Protokolle von *Location Bases Services* (LBS) für Nutzer mobiler Endgeräte, wie z. B. Mobiltelefone, zu standardisieren. LBS sind standortbezogene Dienste, welche positionsabhängige Informationen liefern sollen. Dies findet vor allem bei Informations- und Navigationsdiensten Anwendung. Ein Beispiel wäre eine *Point-of-Interest* (POI) Anfrage, wo der Benutzer eines mobilen Endgerätes mit geographischer Positionsbestimmung eine Abfrage nach z. B. allen Kaffeehäusern in der Umgebung von 500 Meter, machen könnte. Durch die Standardisierung soll Interoperabilität zwischen den einzelnen Netzdiensten geschaffen werden, um eine reibungslose Zusammenarbeit zu ermöglichen. Laut Weiser, Neis und Zipf (2006), definiert OpenLS ein Modell für die Kernkomponenten und Informationspräsentation für Anbieter von *Location Based Services*. Diese Kernkomponenten wurden in den *OpenLS Core Services*

¹⁰<http://www.opengeospatial.org/standards/gml> – zuletzt abgerufen am 03.04.2013.

2 Einführung in die Geoinformationssysteme

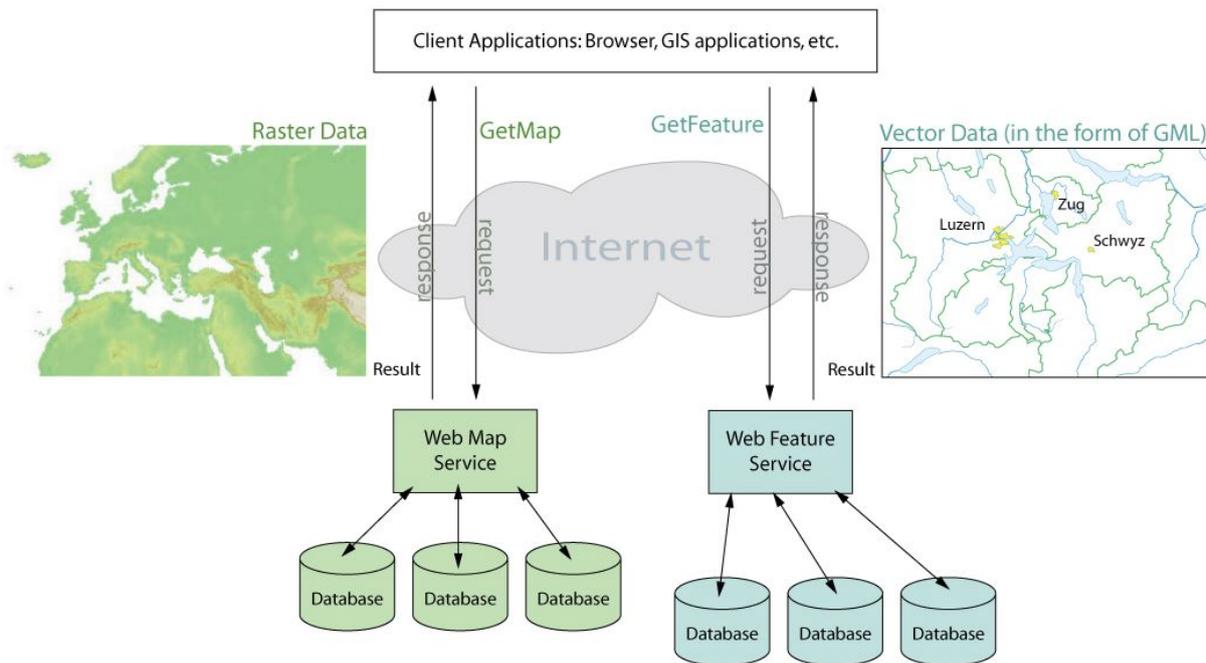


Abbildung 2.2: Der Unterschied zwischen dem Web Map Service und Web Feature Service liegt darin, dass vom WMS Raster- und vom WFS Vektordaten bereit gestellt werden (CartouCHE, 2012).

zusammengefasst. Folgende Dienste gehören laut Mabrouk et al. (2005) zu diesen *OpenLS Core Services*:

- *Directory Service*: Ermöglicht einen Zugriff auf ein Online-Verzeichnis, um nach POI's in der näheren Umgebung zu suchen. Durch OpenLS soll es möglich sein, eine standardisierte Anfrage an einen Dienst zu stellen, welcher dann die Informationen zurücksendet.
- *Gateway Service*: Dieses Service soll die aktuelle Position z. B. über ein Mobiltelefon mit Hilfe von GPS oder im Telefonnetz bestimmen. Es handelt sich hier um die Schnittstelle zwischen GeoMobility Server und Location Server eines Mobilfunknetzbetreibers, sofern diese zur Verfügung gestellt wird.
- *Location Utility Service*: Dieses Service betreibt Geocoding bzw. Reverse Geocoding. Es wird entweder eine Adresse gesendet, die Antwort sind geographische Koordinaten oder es werden geographische Koordinaten gesendet, und als Antwort wird die nächstgelegenen Adresse in der Nähe gesendet.
- *Presentation Service*: Dieses Service ist vergleichbar mit dem WMS und dient der grafischen Präsentation einer Karte.
- *Route Service*: Dieses Service soll der Möglichkeit der Verwendung eines Routenplaners dienen. Die Anfrage beinhaltet Start und Endpunkt, die Antwort wäre dann eine Route zwischen den beiden Punkten.

2.4 Modellierung von geographischen Daten

Geographische Objekte können auf zwei unterschiedliche Arten dargestellt werden: als Vektordaten und als Rasterdaten. Die Vektordaten beschreiben ein Objekt als geometrisches Element, wie Linien oder Flächen. Im Gegensatz dazu bestehen Rasterdaten aus Bildelementen (pixel). Beide Arten der Darstellung haben Vor- und Nachteile, auf die im Folgenden näher eingegangen werden soll.

2.4.1 Vektordaten

Die Darstellung von geographischen Objekten erfolgt bei Vektordaten auf Basis von Koordinatenlisten, welche Punkte enthalten, wobei ein Punkt ein Punktobjekt repräsentiert, zwei bis n Punkte eine Linie, und eine Punktwolke aus zumindest vier Punkten besteht (Jansen & Adams, 2010, S. 30). Diese geometrische Beschreibung eines geographischen Objektes wird als Feature bezeichnet. Zusätzlich werden häufig bei den Vektordaten Attribute angehängt, die Informationen zu den Koordinaten liefern. Dies können z. B. Höheninformationen oder Darstellungsregeln wie Farbe oder Strichstärke der Linien oder Polygonen sein. Zwei durch das OGC definierte gängige Dateiformate für Vektordaten sind das Geographic Markup Language (GML) und das Well-Known Text (WKT).

Geographic Markup Language

Das durch die OGC definierte Datenformat Geographic Markup Language (GML) ist eine im XML Format verwendete Auszeichnungssprache, die zum Austausch raumbezogener Objekte, deren Attribute und Geometrien verwendet wird. GML ist sehr flexibel und erweiterbar, was dem Benutzer erlaubt, Attribute oder Element selbst zu definieren. In diesem Format werden Geodaten, deren Position und deren Attribute, beschrieben. Es wird jedoch nicht vorgeschrieben, wie diese Geodaten präsentiert werden sollen (Peng & Zhang, 2004). Listing 2.1 zeigt einen Auszug einer GML Datei, welche von der Stadt Wien zum Download zur Verfügung gestellt wird¹¹. Es werden die verschiedensten GIS-Daten der Stadt angeboten, unter anderem auch ein Spielplatzverzeichnis. Interpretiert werden diese Informationen als ein geometrisches Punkt-Objekt, definiert durch *gml:Point* und dessen geographische Position mit der Angabe von Längen- und Breitengrad bei *gml:pos*. Zusätzlich hält die GML Datei Information wie z. B. Öffnungszeiten oder in welchem Bezirk sich der Spielplatz befindet. Geometrische Objekte, die mit GML beschrieben werden können, sind jedoch nicht nur auf Punktobjekte begrenzt. Auch komplexere Formen wie Polygone oder LineStrings (eine Sequenz aus miteinander verbundenen Punkten) können über einzelne Punkte definiert werden (Cox, Cuthbert, Lake & Martell, 2001).

¹¹<http://data.wien.gv.at/katalog/spielplaetze.html> – zuletzt abgerufen am 03.04.2013.

```
1 <ogdwien:SPIELPLATZOGD gml:id="SPIELPLATZOGD.64024">
2   <ogdwien:SHAPE>
3     <gml:Point srsDimension="2" srsName="http://www.opengis.net/gml/srs/epsg.
4       xml#4326">
5       <gml:pos>16.450582086047884 48.28681239345763</gml:pos>
6     </gml:Point>
7   </ogdwien:SHAPE>
8   <ogdwien:ANGEBOT> Fussballplatz </ogdwien:ANGEBOT>
9   <ogdwien:OEFFNUNGSZEITEN>0:00-24:00</ogdwien:OEFFNUNGSZEITEN>
10  <ogdwien:STANDORT>Parkanlage Illgasse</ogdwien:STANDORT>
11  <ogdwien:BEZIRK>21</ogdwien:BEZIRK>
12  <ogdwien:WASSER>Nein</ogdwien:WASSER>
13  <ogdwien:WEBLINK1>http://www.wien.gv.at/umwelt/parks/anlagen/index.html
14 </ogdwien:WEBLINK1>
15 </ogdwien:SPIELPLATZOGD>
```

Listing 2.1: Auszug einer von Vienna GIS bereit gestellten GML-Datei (Wien.at, 2013a)

Well-Known Text

Ebenso wie das GML Format wurde Well-Known Text (WKT) von der OGC definiert und eingeführt. Ziel dahinter war es, geometrische Formen im ASCII-Format beschreiben zu können, um diese für Datenbanken zu optimieren und einfach austauschen zu können. Der Aufbau von geometrischen Objekten im WKT Format ist sehr einfach gehalten und für Mensch und Maschine lesbar (siehe Listing 2.2).

```
1 POINT(15 20)
2 LINESTRING(0 0, 10 10, 20 25, 50 60)
3 POLYGON((0 0,10 0,10 10,0 10,0 0),(5 5,7 5,7 7,5 7, 5 5))
```

Listing 2.2: Drei verschiedene geografische Objekte, welche im Well-Known-Text Format beschrieben werden.

Der einfache Aufbau ist bei der POINT- Geometrie zu erkennen. Die x- und y-Koordinaten werden durch ein Leerzeichen getrennt. Soll ein LINESTRING Objekt erzeugt werden, werden die Punkt Koordinaten durch ein Komma getrennt. Der Aufbau eines POLYGON wird durch dessen Grenzlinien definiert. In diesem Beispiel besteht das POLYGON aus einem äußeren und einem inneren Ring. Zu beachten ist ebenso, dass der Anfangs- und Endpunkt identisch sein muss, um ein geschlossenes Polygon zu erzeugen.

2.4.2 Rasterdaten

Im Gegensatz zu Vektordaten werden die Rasterdaten durch Bildelemente in einer Rastermatrix dargestellt (siehe Abbildung 2.3). Diese Bildelemente sind z. B. digitale Luftbilder oder gescannte analoge Karten, die je nach Auflösung aus einer bestimmten Anzahl von Bildpunkten bestehen. Diese Pixel beschreiben die Rasterdaten. In der Literatur spricht man von »simplen« und »klugen« Rasterdaten. Bei den »simplen« besteht ein Pixel lediglich aus dessen Farbwert (z. B. in RGB), bei den klugen Rasterdaten sind noch zusätzliche Informationen pro

2 Einführung in die Geoinformationssysteme

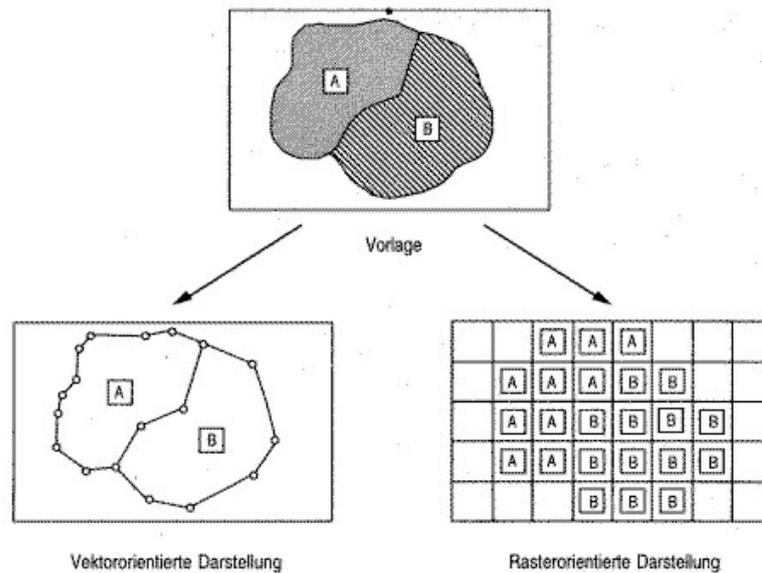


Abbildung 2.3: Unterschied zwischen Raster und Vektordarstellung. Gezeigt wird, wie die Vorlage jeweils entweder durch Vektoren oder Raster dargestellt werden kann. (Cham, 2013)

Pixel gespeichert (z. B. Höhendaten). Verwendet werden Rasterdaten meist zur Darstellung von Luftbildern und Karten als räumliche Bezugsgrundlage.

Georeferenzierte Rasterbilder benötigen einen Referenz im Raum. Dies wird mit einer *World-datei* realisiert, eine Textdatei mit sechs Zeilen, welche alle nötigen Georeferenzdaten der Bild-datei hält. Sie hat meist die selbe Dateibezeichnung wie die Rasterdatei, wobei die Datei-endung an der Worlddatei an die Rasterdatei angepasst wird, z. B. bei *Image.jpeg* wäre die Worlddatei *Image.jpeg*. Den Aufbau einer solchen Datei zeigt Listing 2.3.

```

1  0.05
2  0
3  0
4  -0.05
5  5.025
6  56.975

```

Listing 2.3: Aufbau einer Worlddatei, welche die Position und Lage eines Rasterbildes bestimmt.

In Zeile 1 und 4 wird durch Angabe, wie viele Meter in der Realität ein Pixel entspricht, die Ausdehnung pro Pixel in X- und Y- Richtung bestimmt. Durch Angabe von Werten in Zeile 2 und 3 ist es möglich, die Datei zu rotieren. Die letzte und vorletzte Zeile der Worlddatei geben die Position der Rasterdatei in x- und y-Koordinaten an. Referenzpunkt ist der linke oberste Bildpunkt. Die Werte in Zeile eins bis vier stehen für die x- und y-Komponenten der Pixelhöhe bzw. Pixelbreite. Durch diese zusätzlichen Informationen kann ein GIS mehrere Rasterdaten in korrekter Weise auf der Erdoberfläche kombinieren und darstellen (Jansen & Adams, 2010, S. 36).

2 Einführung in die Geoinformationssysteme

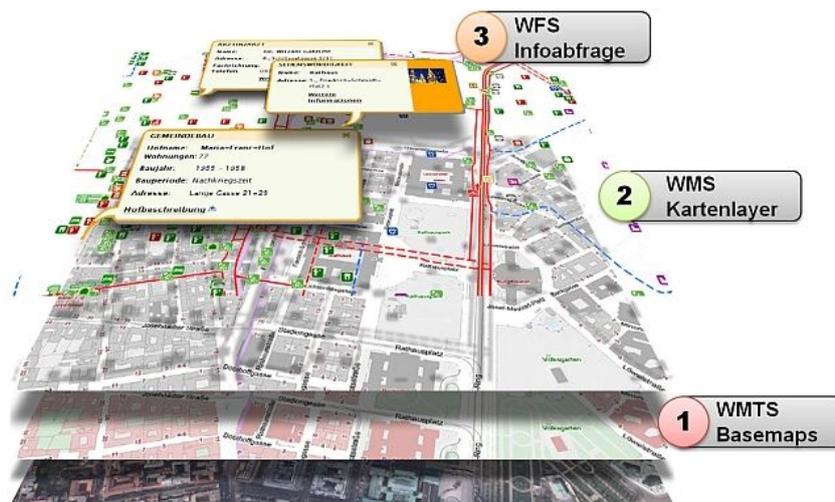


Abbildung 2.4: Aufbau eines hybriden Services. Dieses besteht aus WMTS, WMS und WFS Daten, welche übereinander gelegt eine Karte mit Zusatzinformationen ergeben. (Wien.at, 2013b).

2.4.3 Hybride

Aktuell arbeiten geographische Systeme meist mit hybriden Datenmodellen. Bei einem hybriden (lat. hybridia: Mischling) Datenmodell handelt es sich um die gemeinsame Verwendung von Rasterdaten und Vektordaten. Diese Systeme können beide Datentypen verarbeiten und miteinander in Beziehung setzen, was die Vorteile beider Datenmodelle vereinigen soll. Die Hintergrund- bzw. Basiskarten werden durch Rasterdaten bereitgestellt. Zusatzinformationen werden mit Hilfe von Vektordaten über den Rasterdaten präsentiert. Als Beispiel soll in Abbildung 2.4 der Implementierungsvorschlag für eine Web-Applikation von ViennaGIS gezeigt werden. Als Basiskarte, hier *Orientierungsgrundkarte* genannt, wird eine WMTS (Web Map Tile Service), ein weiterer Geodienst, spezifiziert durch das OGC, präsentiert. Die Karte darüber, welche im Rasterdaten-Format vorliegt, wird mittels WMS eingebunden. Abschließend werden über ein WFS zusätzliche Vektorinformationen dargestellt¹².

2.5 WebGIS

Wird GIS in eine Webanwendung eingebettet, um damit große Mengen an geographischen Daten an eine Vielzahl von Personen im World Wide Web zu verbreiten, wird dies mit WebGIS bezeichnet (Boulos & Honda, 2006). Jiangfeng, Qian, Sen und Xuezhi (2009) beschreiben, dass WebGIS durch die Verwendung eines Standard Internet Protokolls von Nutzern rund um die Welt verwendet werden kann. Es ist plattform-, installations- und ortsunabhängig verfügbar. Immer schneller werdende Internetanschlüsse haben die Entwicklung von Geo-Informationstools im Internet beschleunigt und WebGIS wurde zum wichtigsten Teil von GIS.

¹²<http://data.wien.gv.at/formate/geowebserVICES.html#wfs> – zuletzt abgerufen am 03.04.2013.

2 Einführung in die Geoinformationssysteme

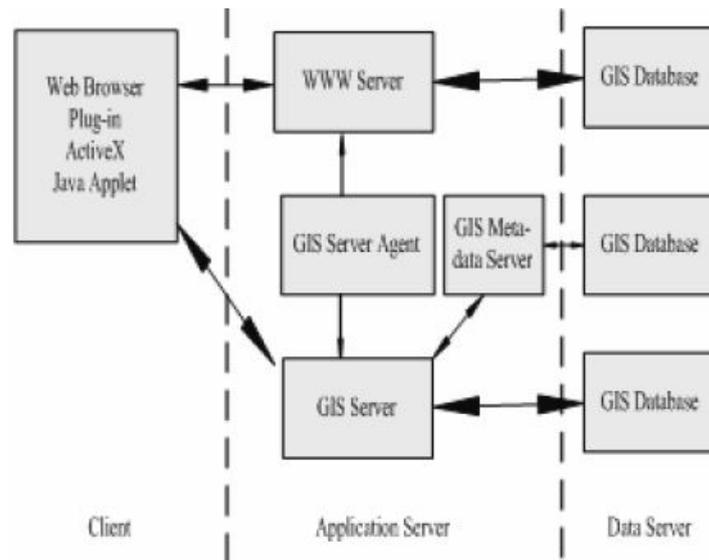


Abbildung 2.5: Darstellung einer WebGIS Architektur bestehend aus einem Drei-Schichten-Modell (Jia & Han, 2010).

Zu den Vorteilen von WebGIS zählen, dass mit Hilfe von gängigen Internetbrowsern Geodaten verbreitet werden und die User jederzeit und überall auf die gewünschten Daten zugreifen können. Dadurch, dass die Standards der Internet Technologie offen sind und durch die Standardisierungsorganisation W3C und IETF¹³ bestimmt werden, führt dies dazu, dass WebGIS leicht erweiterbar und leicht integrierbar ist (Jia & Han, 2010).

WebGIS Architektur

Nach Jia und Han (2010) wird die WebGIS Architektur wie in Abbildung 2.5 als eine Drei-Schichten-Architektur (engl.: *three tier architecture*) dargestellt. Diese Architektur kombiniert Web Technologie mit einer Client-Server Architektur. Die *Client*-Seite besteht typischerweise aus dem Web Browser, auf dem durch Techniken wie z. B. Java Applets, GIS Informationen präsentiert werden. Die *Server*-Seite besteht aus dem *Application Server*, der Logik-Schicht, sowie dem *Data Server* (Datenhaltungsschicht). Die Logik-Schicht übernimmt typischerweise die Verarbeitung und Berechnung der Geodaten, wobei die *Data Server*-Schicht für das Halten und Zurverfügungstellen der Geodaten verantwortlich ist. Diese Architektur verbindet die flexible und einfache Web Architektur mit hoher Sicherheit und Datenintegrität durch den Einsatz von Client- und Server-Seiten.

2.6 Geosemantik

Durch die Definition und Verwendung einheitlicher Schnittstellen ist es möglich, Geodaten effizient zur Verfügung zu stellen und auszutauschen. Auf der Suche nach den gewünschten

¹³<http://www.ietf.org/> – zuletzt abgerufen am 03.04.2013.

2 Einführung in die Geoinformationssysteme

GeoNames Home | Postal Codes | Download / Webservice | About [login](#)

Perth all countries [\[advanced search\]](#)

608 records found for "Perth"

| Name | Country | Feature class | Latitude | Longitude |
|---|---|---|---------------|----------------|
| 1 Perth Gorad Pert, PER, Peairt, Pert, Perta, Pertas, Perth, Perth City, Perthia, Perto, Perp, Pirth, Pirt, Përta, byrth, p... | Australia , Western Australia Cambridge | seat of a first-order administrative division population 1,446,704 | S 31° 57' 8" | E 115° 51' 41" |
| 2 Perth P'rt, PSL, Peairt, Pert, Perth, Перт, Пърт | United Kingdom , Scotland Perth and Kinross | seat of a second-order administrative division population 43,633 | N 56° 23' 42" | W 3° 25' 53" |
| 3 South Perth | Australia , Western Australia South Perth | second-order administrative division | S 31° 59' 55" | E 115° 52' 20" |
| 4 Perth Water | Australia , Western Australia Perth Water | second-order administrative division | S 31° 58' 0" | E 115° 51' 0" |
| 5 Perth / Scone EGPT, PSL | United Kingdom , Scotland Perth and Kinross | airport elevation 121m | N 56° 27' 0" | W 3° 22' 0" |
| 6 Perth Amboy | United States , New Jersey Middlesex County | populated place population 50,814, elevation 17m | N 40° 30' 24" | W 74° 15' 55" |
| 7 South Perth Perth South, South Perth | Australia , Western Australia South Perth | populated place population 11,301 | S 31° 59' 0" | E 115° 52' 0" |

Abbildung 2.6: Die Suche nach *Perth* liefert aufgrund fehlender Zusatzinformationen kein eindeutiges Resultat (GeoNames, 2013).

geographischen Daten kann es jedoch immer noch zu Problemen kommen. Wie in Abbildung 2.6 zu erkennen ist, liefert die Suche nach *Perth* keine eindeutige Antwort zurück. Dies zeigt, dass die Suchanfrage genauer formuliert werden muss. Bei Win und Hla (2005) wird in einem Beispiel beschrieben, wie schwierig es für eine Firma sein kann, nach einer geeigneten Fabrik „in der Nähe einer Autobahn“ zu suchen. Einerseits muss der Suchalgorithmus zunächst verstehen, dass „in der Nähe“ nicht in geographischer räumlicher Nähe bedeutet, sondern erreichbar für PKW oder LKW. Um diese Anfrage korrekt beantworten zu können, sind die nötigen Daten sicherlich vorhanden (Routenplaner, Distanzmessungen, Straßenkarten), jedoch fehlt die maschinenlesbare semantische Interpretation der Suchanfrage.

Die Geosematik befasst sich mit der Aufgabe, mit Hilfe von Techniken und Werkzeugen die Suche nach Informationen zu optimieren. Eine zum Einsatz kommende Technik ist jenes Verfahren, welches auch im semantischen Web eingesetzt wird. Durch das W3C definierte Standards wie RDF¹⁴ oder OWL¹⁵ werden Ontologien gebildet, welche die Semantik von Daten maschinenlesbar machen sollen. Eine weitere Technik, die bei Chen (2007) erläutert wird, sind Microformate¹⁶. Durch dieses Format werden HTML oder XHTML Elemente um semantische Begriffe erweitert. Dadurch gelingt eine genauere Strukturierung von Webseiten, was wiederum zu einer höheren Interoperabilität führt.

Schlussendliches Ziel der Geosematik soll es sein, die bereits vorhandene Menge an Geoinformationen im gleichen Maße für Mensch und Maschine lesbar zu machen. Dies würde ein effizienteres Suchen und Arbeiten mit geographischen Informationen ermöglichen, da nähere semantische Informationen zur Verfügung stehen.

¹⁴<http://www.w3.org/RDF/> – zuletzt abgerufen am 03.04.2013.

¹⁵<http://www.w3.org/TR/owl2-overview/> – zuletzt abgerufen am 03.04.2013.

¹⁶<http://microformats.org/> – zuletzt abgerufen am 03.04.2013.

2.7 Zusammenfassung

Bereits dieses Grundlagenkapitel zeigt, wie allgegenwärtig Geoinformationssysteme geworden sind. Die Forschung und Entwicklung in diesem Bereich bietet immer neuere Möglichkeiten, dem Endbenutzer geographische Informationen effizient und attraktiv zu präsentieren. Durch den Beginn von Standardisierungsbemühungen von geographischen Informationen wurde es immer einfacher, eine Vielzahl verschiedener geographischer Daten miteinander zu kombinieren und auszutauschen. Standards wie das *Web Map Service* oder das *Web Feature Service* ermöglichen es, Informationen aus den verschiedensten Quellen in einer Anwendung zu integrieren. Durch *Hybride* Präsentation können die eingebundenen Karten und Geodaten übereinander gelegt und effektiv dargestellt werden. Diese Fakten in Verbindung mit WebGIS (ortsunabhängige Verwendung von GIS Software im World Wide Web) erlauben eine Vielzahl von neuen und innovativen Anwendungen. Diese neuen Anwendungen werden sich auch mit dem Problem der Geosemantik auseinandersetzen müssen, um die Datenverbreitung durch semantische Informationen effektiver gestalten zu können.

In dem nachfolgenden Kapitel werden ausgewählte Themen mit Bezug auf Geoinformationen die umfassenden Möglichkeiten von GIS näher beschreiben. Zu diesen Themen zählen bereits erwähnte Tools wie Routenplaner oder Satellitennavigationssysteme und deren Techniken, sowie die Verwendung von proprietären oder freien Diensten. Daraus soll hervorgehen, wie Geodaten erstellt, weiter verarbeitet und anschließend präsentiert werden können.

3 Ausgewählte Themen

In diesem Kapitel wird auf ausgewählte Themen in Bezug auf Geoinformationen eingegangen. Die Entwicklung von Positionierungssystemen zeigt, wie es möglich ist, Geodaten zu bestimmen und aufzuzeichnen. Im World Wide Web werden diese aufgezeichneten Geodaten von verschiedenen Portalen zur weiteren Verwendung angeboten. Diese Daten werden durch proprietäre, wie bei Google Maps oder Bing Maps, oder freie Dienste, wie bei OpenStreet-Maps, angeboten. Des Weiteren wird näher auf die Motivation hinter dem OpenStreetMap Crowdsourcing Projekt eingegangen und dessen zusätzliche Funktionen und deren Handhabung im Kurzen erläutert. Dies zeigt, wie es mit freien Alternativen möglich ist, die gleiche Funktionalität wie bei proprietären Diensten erhalten zu können.

3.1 Globale Navigationssatellitensysteme

Als globale Navigationssatellitensysteme, auch GNSS (engl. *Global Navigation Satellite System*) genannt, werden Systeme bezeichnet, die Positionsbestimmung und Navigation durch Signale von Navigationssatelliten ermöglichen. Da GNSS Empfangsgeräte in den letzten Jahren immer kostengünstiger geworden sind, finden sie in vielen unterschiedlichen Bereichen Anwendung. Da der Einsatz von GNSS - im Kontext der Arbeit - vielfach bei Outdoor Sport Aktivitäten Verwendung findet, wird dieser Bereich genauer analysiert.

3.1.1 Entwicklung von Satellitennavigation

Den Beginn der Entwicklung von Satellitennavigation kann laut Thiele (2008, S. 4) auf den Tag datiert werden, als die Sowjetunion am 4. Oktober 1957 den ersten Satelliten der Welt („Sputnik“) ins All brachte. Auch in den USA bestand großes Interesse an dem Satelliten und dessen Flugdaten. Mitten im Kalten Krieg war jedoch ein Austausch von Informationen zwischen den USA und der UdSSR nicht möglich, daher musste man sich andere Wege überlegen. Durch Triangulierung des vom Satelliten abgestrahlten Signals mit drei Bodenstationen konnte die Flugbahn berechnet werden. Daraus erkannte man den Umkehrschluss, dass es auch mit drei Satelliten möglich sein müsste, eine unbekannte Koordinate eines Punktes auf der Erde berechnen zu können. Durch das Militär in Auftrag gegeben, entstand in den 90er Jahren das erste satellitengestützte Navigationssystem *GPS* (Global Positioning System).

Ursprünglich konnte *GPS* nur vom Militär genutzt werden, bis zu einer folgenschweren Tragödie im September 1983. Ein Passagierflugzeug der Korea Airlines geriet durch einen Navigationsfehler in den sowjetischen Luftraum und wurde abgeschossen. Daraufhin schlug der damalige US-Präsident Reagan vor, *GPS* auch für die zivile Nutzung freizugeben. Bis ins Jahr 2000 wurde *GPS* durch die „selektiven Verfügbarkeit“ (Selective Availability, *SA*) beeinflusst. Diese Technik ermöglichte nur dem US-Militär ein exaktes *GPS* Signal zu empfangen,

3 Ausgewählte Themen

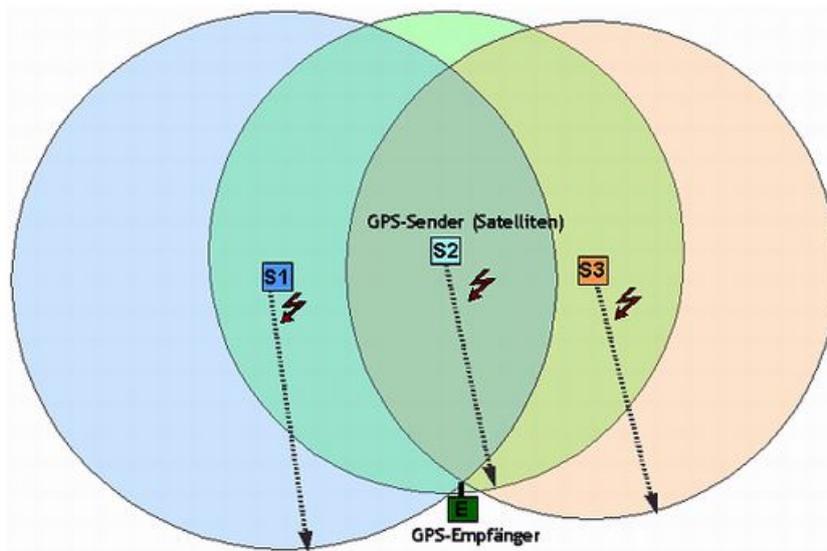


Abbildung 3.1: Positionsbestimmung eines GPS-Empfängers mit Hilfe von drei Satelliten (S1, S2, S3) (Nathansen, 2013).

wo hingegen für zivile Nutzer ein großer Fehler in der Positionsbestimmung enthalten war. Dadurch konnte die Position nur auf plus/minus 100 Meter bestimmt werden. Dies hatte den Sinn, dass Staatsfeinde oder Terroristen das System nicht für punktgenaue Raketenangriffe nutzen sollten. Im Jahr 2000 wurde die SA deaktiviert, kann bei Bedrohung jedoch jederzeit wieder aktiviert werden (Haklay & Weber, 2008).

Parallel zum amerikanischen GPS entwickelte die Sowjetunion *GLONASS*, ebenso ein globales Satellitennavigationssystem, das jedoch von Anfang an auch für die zivile Nutzung verfügbar war (Thiele, 2008, S. 4). In Europa wird mit *Galileo* ein Navigationssystem aufgebaut, welches 2020 voll betriebsfähig sein soll (Genesys, 2013).

3.1.2 Funktionsweise der Satellitenortung

Für eine Positionsbestimmung auf der Erde werden die Signale von mindestens vier Satelliten benötigt. Diese Signale senden Informationen über die Koordinaten des Satelliten und die Uhrzeit ihrer Abstrahlung. Durch Kenntnis der Uhrzeit am Empfänger kann die Differenz berechnet werden, woraus sich eine Dauer ergibt, die das Signal vom Satelliten zum Empfänger benötigt. Aus dieser Dauer lässt sich durch Multiplikation der Lichtgeschwindigkeit die Distanz berechnen. Wie in Abbildung 3.1 zu sehen, bildet diese Distanz den Radius einer Kugel, wobei der Satellit den Mittelpunkt, und die Position des Empfängers einen Punkt am Mantel der Kugel darstellt. Dieser Vorgang wird mit drei Satelliten durchgeführt, wodurch eine Triangulierung der Position ermöglicht wird (Thiele, 2008, S. 5).

Allein durch diese Triangulierung von drei Satelliten wäre eine Positionsbestimmung möglich, allerdings ist diese Position wegen des „Zeitfehlers“ sehr ungenau. Wie bei Bajaj, Ranaweera und Agrawal (2002) beschrieben, kann eine ungenaue Zeit am Empfänger von einigen Millisekunden Schuld an einer falsche Position von einigen Kilometern sein. Dadurch wird zur

3 Ausgewählte Themen

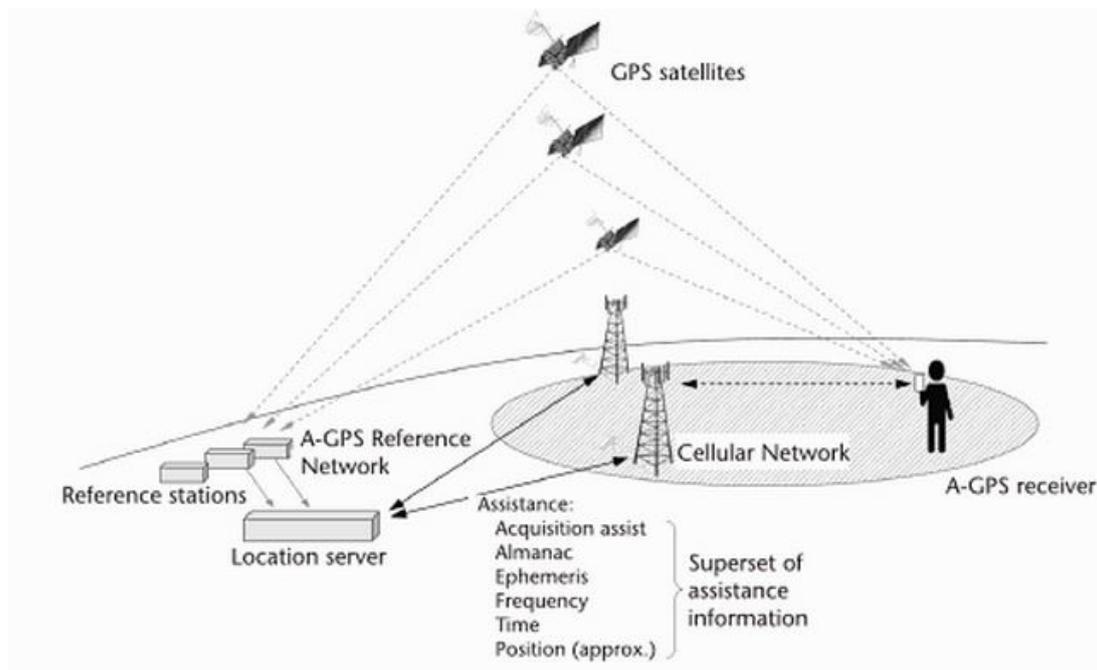


Abbildung 3.2: A-GPS bestimmt die Position durch die Verbindung von Satellitennavigation und Mobilfunktechniken. (Van Diggelen, 2009, S. 3)

Kontrolle ein vierter Satellit in die Berechnung einbezogen und so eine exakte Positionsbestimmung ermöglicht.

In der Theorie kann durch den beschriebenen Vorgang eine exakte Bestimmung der Position durchgeführt werden, in der Praxis ist dies jedoch nicht der Fall. Die Dauer, die das Signal vom Satellit zum Empfänger benötigt, kann durch atmosphärische Wasserpartikel beeinflusst werden. Eine weitere Fehlerquelle ist die Anzahl und Position der Satelliten. Je mehr Satelliten empfangen werden können, desto genauer wird die Position berechnet. Bei lediglich vier Satelliten, die einen schleifenden Schnittpunkt bilden, wird es zu einem Fehler in der Positionsbestimmung kommen (Bajaj et al., 2002). Durch Beachtung aller möglichen Störungsquellen resultiert ein Fehler von plus/minus 12 Meter. Neue Techniken wie WAAS oder EGNOS versuchen den Fehler durch den Einfluss der Atmosphäre zu reduzieren. Dadurch kann eine Genauigkeit der Positionsbestimmung von plus/minus 3-5 Meter erreicht werden (Kowoma, 2008).

3.1.3 Einsatzbereiche von Satellitennavigation

Vor allem im Transportwesen findet der Einsatz von Satellitennavigation zur Positionsbestimmung statt, wie z. B. in der Luftfahrt, der Seefahrt oder bei Kraftfahrzeugen. Für diese Arbeit interessante Einsatzgebiete sind auch Outdoor Sportaktivitäten. Hier kommen einerseits GPS Empfänger zum Einsatz, welche z. B. als Uhr am Handgelenk getragen werden können, andererseits wird das Smartphone durch die Integration von GPS eingesetzt. Ein Vorteil von GPS in Smartphones ist die Verwendung von *Assisted Global Positioning System (A-GPS)*.

Geräte, welche mit der *A-GPS*-Technik ausgestattet sind, haben die Möglichkeit, durch die Nutzung zusätzlicher Verfahren die Positionsbestimmung zu beschleunigen und durch die Kombination mit der Satellitenortung ein genaueres Resultat zu ermöglichen (Van Diggelen, 2009, S. 3). Befindet sich das Smartphone in einem Mobilfunknetz, so kann mit Hilfe der Sendemasten eine zusätzliche Positionsbestimmung erreicht werden (siehe Abbildung 3.2).

GPS-Empfangsgeräte bieten die Möglichkeit, die aktuellen Positionen als Wegpunkt mit Breitengrad, Längengrad und Zeit zu speichern. Aus diesen Positionen ergibt sich dann eine Route. Durch Analyse dieser Routen können die Entfernung, Höhenmeter und Geschwindigkeit bestimmt werden, die eine exakte Auswertung der erbrachten sportlichen Leistung erlauben. Die Berechnung und Auswertung einer Route, gespeichert in einer GPX-Datei, wird im praktischen Teil dieser Arbeit erläutert.

3.2 Proprietäre und freie Geodaten

Der nächste Abschnitt befasst sich mit unterschiedlichen Diensten, welche Geodaten im Web, wie Kartenmaterial oder Geoservices, anbieten. Hier ist es möglich, das Angebot in zwei Bereiche zu unterteilen: Zum einen *proprietäre* Geodaten, zum anderen *freie* Geodaten. Worin genau der Unterschied liegt, und warum auch proprietäre Geodaten zunächst frei im Zugang und Verwendung wirken, wird im Folgenden genauer beschrieben.

Mit dem Begriff *proprietär* kann jene Software bezeichnet werden, die zwar frei verfügbar ist, jedoch weder verändert werden kann, noch ist der Source Code ersichtlich. Es handelt sich also um geschlossene Software, die nur genutzt werden kann, wie sie vorliegt (Rostock, 2012). Der Grund, proprietäre Software bereit zu stellen, wurde in dieser Arbeit bereits im Grundlagenkapitel genannt: Geodaten sind sehr teuer in ihrer Erstellung und Wartung. Um die Technologien, Komponenten und Daten zu schützen, wird geschlossene, nicht einsehbare Software gewählt.

Trotz der hohen Kosten in der Erstellung gibt es Firmen, die ihre Geodaten-Software scheinbar frei zur Verfügung stellen. Die in diesem Abschnitt näher betrachteten Dienste von Google Maps und Bing Maps sind über Webbrowser frei verwendbar. Des Weiteren bieten alle drei eine API Schnittstelle um Anwendungen oder Webclients der angebotenen Dienste zur Verwendung bereit zu stellen. Das „scheinbar frei“ bezieht sich auf die umfangreichen Nutzungsbedingungen, die durch das Verwenden dieser Dienste eingegangen werden. In diesen Bedingungen nehmen sich die Firmen verschiedene Rechte heraus, wie das Ändern der API oder das Anzeigen von Werbung über der Karte. Da sich die angebotenen Dienste in vielen Punkten überschneiden, wird ein Auszug aus den API's von Google Maps und dessen Funktionsumfang gegeben. In weiterer Folge wird kurz auf die Nutzungsbedingungen der einzelnen Firmen eingegangen.

3.2.1 Google Maps

Folgende Webdienste werden von der Google Maps API¹ angeboten:

- *Directions API*: Mit Hilfe einer HTTP-Anfrage berechnet die Directions API eine Route zwischen zwei Punkten. Die Routenberechnung kann für verschiedene Transportmittel optimiert werden, einschließlich öffentlicher Verkehrsmittel (nicht weltweit verfügbar), Kraftfahrzeuge, zu Fuß oder mit dem Fahrrad. Die Standorte können als Straßennamen, Ortsnamen, als Textzeichenfolge, oder als Längen- und Breitengrad angegeben werden.
- *Distance Matrix API*: Dieser Dienst berechnet die Entfernung und Fahrzeit für eine Matrix aus Start- und Zielorten. Im Gegensatz zur Directions API werden hier keine Routeninformationen zurückgegeben, lediglich die Werte *duration* und *distance*.
- *Elevation API*: Hier werden Höheninformationen für alle Standorte auf der Erdoberfläche zurückgegeben. Laut Definition dieses Dienstes wird bei Nichtvorhandensein einer exakten Höheninformation der angegebenen Koordinaten eine Interpolation der nächstgelegenen Punkte durchgeführt.
- *Geocoding API*: Hier werden Adressen in geografische Koordinaten umgewandelt und zurückgegeben, et vice versa.
- *Places API*: Dieser Dienst bietet die Möglichkeit, durch Angabe von Koordinaten interessante Orte in der Nähe (*Points of Interest POI*), wie Restaurants oder Bars, anzufordern. Durch die Angabe zusätzlicher Parameter kann die Anfrage weiter eingeschränkt werden, sowie der Radius in dem gesucht werden soll.

Google Maps Nutzungsbedingungen

Die Nutzung der Google Maps API ist zunächst auf eine gewisse Anzahl von „Anfragen pro Tag“ limitiert (Google, 2012b). So darf der Routenplaner, welcher über die Directions API angefordert wird, 2500 mal pro Tag angefragt werden. Dieses Service ist kostenlos, sollte jedoch die Limitierung den Benutzer einschränken, ist ein kostenpflichtiger Zugang zur API möglich (Google Maps API für Unternehmen). Hier steigt das Limit der „Anfrage pro Tag“ für den Routenplaner auf 100.000. Zusätzlich müssen auch die eingesetzten Google Dienste für den Endnutzer kostenlos und öffentlich zugänglich sein. Als weiteren Punkt nimmt sich Google heraus, die API jederzeit zu ändern oder auch Werbung auf den Karten zu platzieren. Zusätzlich ist das Verwenden von erhaltenen Informationen aus der API auf anderen Karten wie Google Karten verboten.

Bing Maps Nutzungsbedingungen

Die Bing Maps Nutzungsbedingungen sind sehr umfangreich. Die Verwendung der Services ist in einige Kategorien eingeteilt, wie z. B. die Verwendung lediglich zu Ausbildungszwecken, für Nicht-Kommerzielle, für Kommerzielle, für mobile Applikationen oder für Windows Applikationen. Im Falle dieser Arbeit treffen die Nutzungsbedingungen des Abschnitts „Nur für Ausbildungszwecke und Non-Profit Organisation“ zu. Auch hier stößt man auf einen mehrere Seiten langen Kontrakt, der wiederum die Verwendung der API's limitiert.

¹<https://developers.google.com/maps/documentation/webservices/> – zuletzt abgerufen am 26.04.2013.

Die Dienste von Google, Bing oder anderen Firmen, die Geodaten anbieten, werden in vielen Fällen für die Grundbedürfnisse der Kunden ausreichen. Es können Karten angezeigt, Routeninformationen dargestellt oder nach interessanten Punkten in der Umgebung gesucht werden. Schnell wird jedoch klar, dass durch die strengen Lizenzbestimmungen ein effektives Arbeiten in vielen Bereichen nicht möglich ist. Aus diesem Bedürfnis heraus entstanden Projekte und Plattformen die als *Volunteered Geographic Information (VGI)* bezeichnet werden, dem freiwilligen Bereitstellen von geografischen Informationen. Als die bekanntesten Vertreter werden Wikimapia², Google Map Maker³ und OpenStreetMap⁴ genannt (Goodchild, 2007).

Bei Wikimapia handelt es sich um eine Plattform, bei der versucht wird, eine geografische Enzyklopädie aufzubauen. Dem User wird eine Karte präsentiert und durch Klicken auf einen Bereich bekommt man Informationen zu diesem, wie z. B. durch das Auswählen der Stadt *Graz* werden Screenshots und eine kurze Beschreibung bzw. die Einwohnerzahl der Stadt gegeben. Diese Informationen sollen wie bei Wikipedia von den Usern selbst erstellt werden.

Google Maps versucht durch den relativ neuen Dienst „Google Maps Maker“ (Start war im Jahr 2008) die Vorteile von *VGI* auch auf ihre Karten anzuwenden. User können nach einer Registrierung geografische Informationen wie Straßen, Flüsse, Bahnstrecken, Gebäudeumrisse und vieles mehr zu den Karten von Google Maps hinzufügen. Auf der Übersichtskarte von Google Maps Maker ist zu erkennen, dass viele europäische Länder noch nicht zum Bearbeiten freigeschaltet wurden, wie z. B. Italien oder Deutschland. Das zeigt, dass sich dieser Dienst noch im Aufbau befindet. Da OpenStreetMaps einen wichtigen Teil dieser Masterarbeit darstellt, wird dieses Projekt im folgenden Abschnitt näher und ausführlicher als Wikimapia und Google Maps Maker beschrieben.

3.2.2 OpenStreetMaps

OpenStreetMap (OSM) ist eine Open Data Plattform, in welcher Geodaten von Benutzern generierten Karten bereitgestellt werden. Begonnen hat die Entwicklung von OSM bereits nach der Deaktivierung der „Selectiven Verfügbarkeit“ des GPS im Jahr 2000. Dadurch wurde es möglich, eine Genauigkeit bei der Positionsbestimmung von 6-10 Meter zu erreichen, was zuvor nur auf 100 Meter genau möglich war. Zusätzlich zu dem immer billiger werdenden GPS Empfängern, besseren Heimcomputern und schnelleren Zugang zum Internet, war es nun für jedermann möglich, aufgezeichnete Geoinformationen zu bearbeiten und zu verbreiten. Daraus entwickelten sich eine Reihe von Projekten, die sich mit dem Austausch dieser Geodaten befassten, von denen das OpenStreetMap Projekt als das umfangreichste und aktuellste gilt (Haklay & Weber, 2008).

Das OSM Projekt wird bei Haklay und Weber (2008) ebenso wie Wikipedia als Resultat von „Crowdsourcing“ gesehen, ein Begriff, der sich vom »outsourcing« ableitet. Als Outsourcing bezeichnet man in der Ökonomie das Auslagern von Produktionsstätten in günstigere Bereiche der Erde, um so billiger produzieren zu können und einen höheren Ertrag zu erhalten (Gabler, 2013). Daraus entwickelte sich das »crowdsourcing« welches laut Definition „[...] das Auslagern von bisher in einem Unternehmen selbst erbrachten Leistungen auf eine große Anzahl von Menschen über das Internet.“ ist (Duden, 2013). Dies findet vor allem in Gebieten Anwendung,

²<http://wikimapia.org/> – zuletzt abgerufen am 03.04.2013.

³<http://www.google.com/mapmaker?hl=de> – zuletzt abgerufen am 03.04.2013.

⁴<http://www.OpenStreetMap.org> – zuletzt abgerufen am 03.04.2013.

3 Ausgewählte Themen

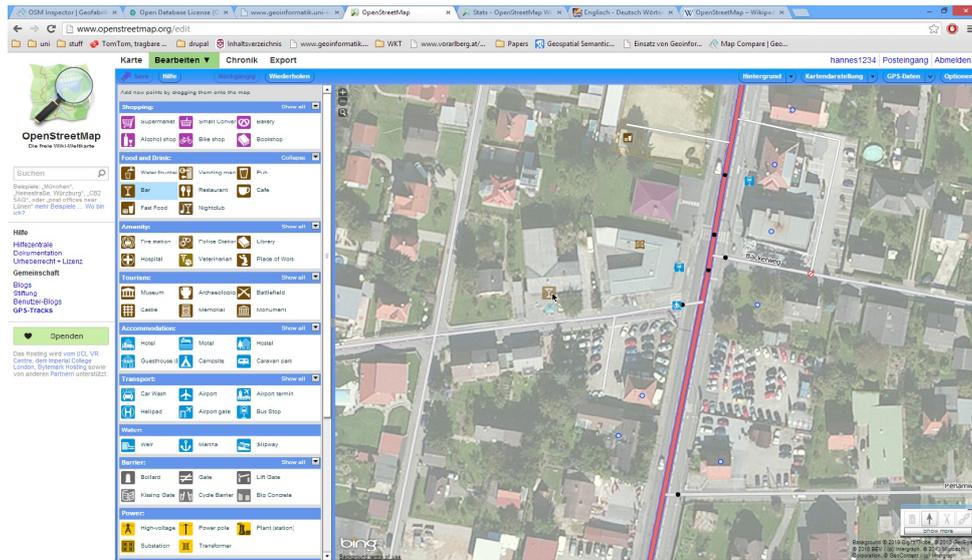


Abbildung 3.3: Mit Hilfe eines Online-Tools lassen sich Informationen zur OSM-Karte hinzufügen. In diesem Beispiel wird eine Bar auf der Karte markiert (OpenStreetMap, 2013b).

die entweder sehr kostenintensiv in der Erstellung sind, oder nicht oder nur schwer automatisiert erstellt werden können. Bei OSM, welches 2004 von Steve Coast gegründet wurde, gelingt es durch die Arbeit vieler freiwilliger Unterstützer, Geodaten zu erstellen, welche „[...] frei zum Kopieren, Weitergeben, zum Übermitteln sowie Anzupassen, sofern OpenStreetMap und die Mitwirkenden als Quelle angegeben werden [...]“ sind (OpenStreetMap, 2013c).

Erstellen und Bearbeiten der Daten

Der Inhalt und die Daten von OSM besteht aus Beiträgen von Freiwilligen. Ähnlich wie bei einem Wiki (Wikipedia) kann jeder Benutzer, nach einer kostenlosen Registrierung, Inhalte hinzufügen oder bestehende Eintragungen ändern. Bei Haklay und Weber (2008) werden unterschiedliche Motivatoren angegeben, warum Mitwirkende ohne eine Vergütung sich Zeit nehmen, Inhalte zu OSM hinzuzufügen. Zum einen sind es ideologische Gründe, der Welt freie Geodaten zur Verfügung zu stellen, zum anderen ist es einfach die Freude daran, hinaus zu gehen, Geodaten zu sammeln, und diese dann mit einer Community zu teilen. Als wichtiger Punkt wird auch genannt, dass die soziale Verbindung unter den Mitgliedern bei regelmäßig veranstalteten „Mapping Parties“ gestärkt wird. Bei diesen Events sollen vor allem neuen Mitgliedern bei den ersten Schritten geholfen und die Bindung innerhalb der Community gestärkt werden.

Die Geodaten können direkt in einem Webbrowser bearbeitet bzw. Straßen oder Wege können durch Upload einer GPX-Datei erstellt werden. Je mehr unterschiedliche GPX-Dateien einen Weg definieren, desto genauer kann dieser bestimmt werden. Ebenso können Cafés, Bars, Haltestellen, Parkplätze, etc. angegeben werden. In Abbildung 3.3 ist zu sehen, wie durch einfaches Platzieren des „Bar“ Logos an der richtigen Position der Karte diese als eine solche definiert wird. Zusätzliche können noch Attribute, wie der Name oder die genau Adresse, angegeben werden.

3 Ausgewählte Themen

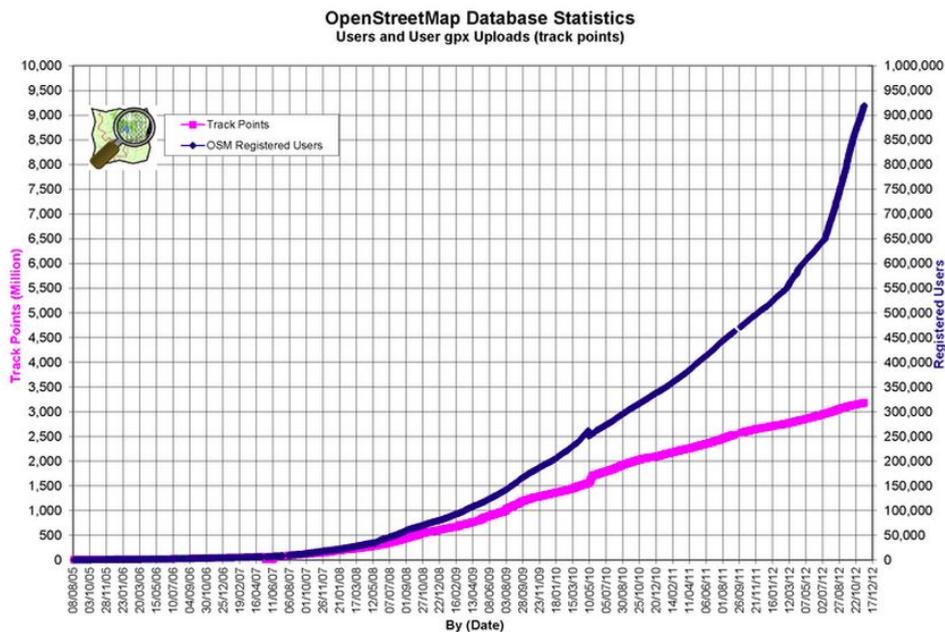


Abbildung 3.4: Diese Statistik zeigt den Anstieg von Wegpunkten und registrierten Nutzern von OpenStreetMap.

Das Diagramm in Abbildung 3.4 zeigt, dass es momentan beinahe 950.000 registrierte User bei OSM gibt. Die zweite Kurve gibt die Anzahl der Wegpunkte im System an. Dass diese Kurve nicht ähnlich anwächst wie die der Benutzer, zeigt, dass die vorhandenen Daten durch die Kontrolle bzw. Updates vieler User immer wieder gewartet und verbessert werden.

Qualitätssicherung der OpenStreetMap Daten

Wie bei allen Plattformen, bei denen diverse Inhalte von Usern hinzugefügt werden, stellt sich die Frage nach der Qualität und Aktualität dieser Daten. In erster Linie versucht OSM durch die Zusammenarbeit der Mitglieder in einem geografischen Gebiet Fehler zu erkennen und zu entfernen. In großen Ballungszentren stellt diese Vorgehensweise keine Probleme dar, da genug Benutzer vorhanden sind, jedoch wird die Erstellung von Geodaten in abgelegenen Gebieten schwierig. Zusätzlich wurden Tools entwickelt, die Fehler im Kartenmaterial aufspüren und melden bzw. selbst beheben. Hierzu zählt, wenn z. B. zwei oder mehrere Straßen an einer Kreuzung nicht verbunden sind wird diese Verbindung von der Software automatisch erstellt. Zusätzlich ist es jedem, der eine OpenStreetMap Karte betrachtet und Fehler findet, möglich, diese zu melden, ohne sich dafür anmelden zu müssen.

Durch die große Anzahl an Beteiligten bieten die OSM Karten in vielen Bereichen eine größere Informationsdichte an. Abbildung 3.5 zeigt einen direkten Vergleich der Karten von OpenStreetMap und Google Maps. Der Bereich bezieht sich auf den Bezirk Eggenberg im Westen von Graz und das Naherholungsgebiet Plabutsch. Gut zu erkennen ist die größerer Dichte an Wegen auf der OSM Karte, welche wiederum in unterschiedliche Klassifizierungen unterteilt werden (Fusswege, Radwege).

3 Ausgewählte Themen

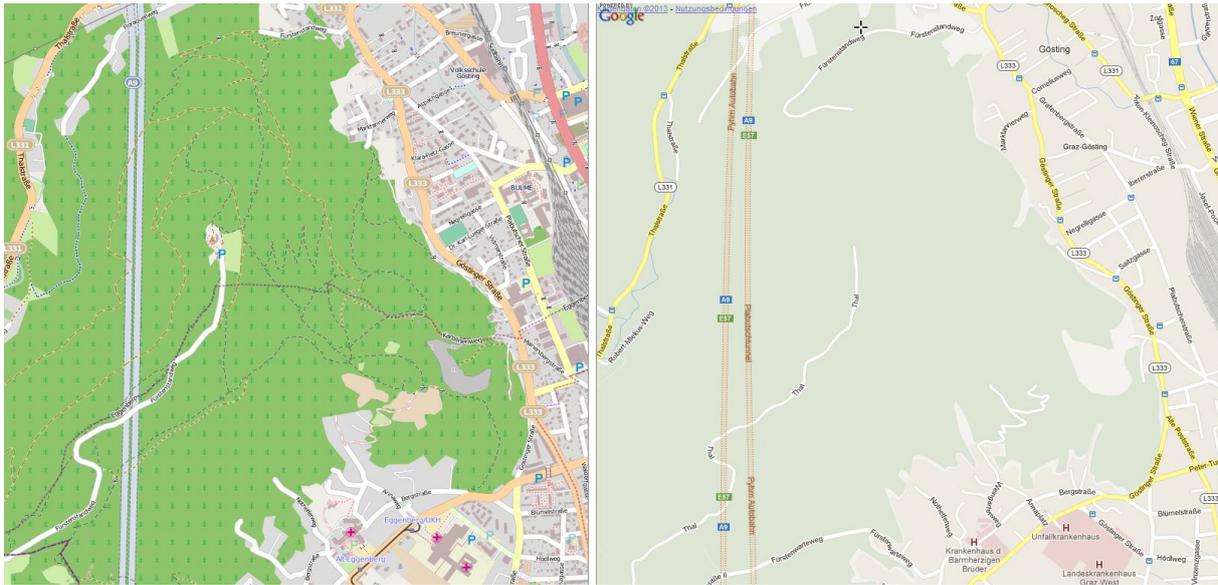


Abbildung 3.5: Vergleich von OSM (links) zu Google Maps (rechts). Zu erkennen ist, dass auf der Google Karte keinerlei Wanderwege eingezeichnet sind.

3.2.3 Freie Geocoding Dienste

Dienste, welche von der Google Maps API angeboten werden, können auch durch frei zugängliche Services abgedeckt werden. Der Unterschied liegt meist darin, dass nicht alle Dienste von derselben Plattform in einer Art abgedeckt werden, wie es bei Google oder Bing der Fall ist. So werden Geocoding oder Point-of-Interest Dienste direkt von OpenStreetMap angeboten, Routenplaner oder die Abfrage von Höhendaten müssen aber von anderen Seiten in Anspruch genommen werden. Zunächst wird der Geocoding Dienst *Nominatim* und die Möglichkeit, nach Point-of-Interest in der Nähe eines Punktes zu suchen, näher beschrieben.

Nominatim

Nominatim ist ein Werkzeug, welches erlaubt, in der OSM Datenbank nach Orten zu suchen. Es besteht die Möglichkeit, das Werkzeug herunterzuladen und lokal auf einem Server zu installieren, es wird jedoch auch eine API Schnittstelle als Zugriff angeboten. Die API muss in folgender Form verwendet werden:

```
1 http://nominatim.openstreetmap.org/search/<query>?<params>
```

Listing 3.1: Aufbau der API für eine Nominatim Anfrage

Der Suchbegriff wird durch *query* definiert. Hier können einzelne Orte wie Straßennamen, Hausnummern, Flughäfen oder Bahnhöfe abgefragt werden. Eine gültige Abfrage wäre z. B. Graz+Hauptbahnhof (die einzelnen Begriffe müssen mit einem + verbunden werden). Die *params* können optional dazu angegeben werden. Hier wird definiert, welche und auf welche Art die Informationen zurückgegeben werden (z. B. Definieren des Ausgabeformats als HTML, XML oder JSON, eine Box kann definiert werden, in der nach dem Ergebnis gesucht wird,

3 Ausgewählte Themen

```
1 <reversegeocode timestamp="Mon, 27 May 13 12:22:28 +0000" attribution="Data
  OpenStreetMap contributors, ODbL 1.0. http://www.openstreetmap.org/
  copyright" querystring="format=xml&lat=47.07197&lon=15.4423&zoom=18&
  addressdetails=1">
2   <result place_id="60398351" osm_type="way" osm_id="84579393" ref="Grazer Dom"
     lat="47.07197935" lon="15.442356530551"> Grazer Dom, 3, Burggasse,
     Innere Stadt, Graz, Steiermark, 8010, Oesterreich, Europaeischen Union
3   </result>
4   <addressparts>
5     <place_of_worship>Grazer Dom</place_of_worship>
6     <house_number>3</house_number>
7     <road>Burggasse</road>
8     <city_district>Innere Stadt</city_district>
9     <city>Graz</city>
10    <county>Graz</county>
11    <state>Steiermark</state>
12    <postcode>8010</postcode>
13    <country>Oesterreich</country>
14    <country_code>at</country_code>
15    <continent>Europaeischen Union</continent>
16  </addressparts>
17 </reversegeocode>
```

Listing 3.2: Die XML Rückgabe einer Nominatim Anfrage zeigt die Vielzahl an Informationen, welche für ein Koordinatenpaar vorhanden sind.

um Mehrfachresultate zu vermeiden). Die zurückgegebenen Informationen beinhalten unter anderem zusätzliche Ortsangaben, wie das Land in dem der Ort liegt sowie der Ländercode, und des Weiteren auch Längen- und Breitengrad des Ortes.

Sollten Längen- und Breitengrad vorhanden und nähere Information zu dieser Position gewünscht sein, werden die Koordinaten mit einem führenden *reverse* übergeben. Hier ist es möglich, bei den Parametern anzugeben, wie genau das gewünschte Ergebnis aussehen soll. Der Umfang der zurückgegebenen Informationen kann so von „nur dem Land“, in dem sich der Wegpunkt befindet, bis hin zur genauen Adresse (sofern vorhanden) erstrecken. Zur besseren Veranschaulichung sei auf Listing 3.2 hingewiesen, welche die XML Antwort auf eine API Anfrage mit Längen und Breitengrad ergibt. Die Abfragesyntax sieht wie folgt aus:

```
1 http://nominatim.openstreetmap.org/reverse?format=xml&lat=47.07197&lon=15.4423&
  zoom=18&addressdetails=1
```

Listing 3.3: Aufbau der API für eine Reverse-Nominatim Anfrage

Extended API

Zusätzlich zur Nominatim bietet OSM eine *Extended API (XAPI)*. Es handelt sich hier um eine modifizierte Version der Standard API, die ebenfalls wie Nominatim auf die OSM Datenbank

3 Ausgewählte Themen

```
1 <osm version="0.6" generator="Osmosis SNAPSHOT-r26564">
2   <node id="287375044" version="4" timestamp="2010-11-19T17:44:43Z" uid="9605"
3     user="a_uller" changeset="6408770" lat="47.0429639" lon="15.3958224">
4     <tag k="public_transport" v="platform"/>
5     <tag k="shelter" v="no"/>
6     <tag k="highway" v="bus_stop"/>
7     <tag k="bench" v="yes"/>
8     <tag k="name" v="Gablenz-Kaserne"/>
9     <tag k="is_in" v="Graz,Graz,Steiermark,Oesterreich,Europe"/>
10    <tag k="bin" v="yes"/>
11  </node>
12  <node id="287375046" version="5" timestamp="2010-12-07T15:39:43Z" uid="9605"
13    user="a_uller" changeset="6576105" lat="47.0425912" lon="15.3956292">
14    <tag k="public_transport" v="platform"/>
15    <tag k="shelter" v="no"/>
16    <tag k="highway" v="bus_stop"/>
17    <tag k="bench" v="yes"/>
18    <tag k="name" v="Gablenz-Kaserne"/>
19    <tag k="is_in" v="Graz,Graz,Steiermark,Oesterreich,Europe"/>
20    <tag k="bin" v="yes"/>
21  </node>
22 </osm>
```

Listing 3.5: XML Rückgabe einer Nominatim Anfrage

zugreift. In der Anfrage muss zunächst eine Box definiert werden, welcher geografische Bereich untersucht werden soll. Zusätzlich kann angegeben werden, nach welchen Elementen gesucht werden soll, z. B. eine *node* oder ein *way*. Zusätzlich werden die Attribut-Prädikate angegeben, die als Map Features definiert sind.

In der folgenden beispielhaften Anfrage (siehe Listing 3.4) wird nach allen Bushaltestellen in einem definierten Bereich gesucht. Die Haltestelle trägt laut Map Features den *Key* *highway* und die *value* *bus stop*. Listing 3.5 zeigt die Rückgabeinformationen dieser Anfrage in XML Format. Zwei *node* Elemente geben die Anzahl der gefundenen Haltestellen im Bereich an und halten Daten wie Längen- und Breitengrad der *Node*. Mit *tag* Elementen werden zusätzliche Informationen zu den einzelnen Nodes zurückgegeben (z. B. der Name der Haltestelle und ob ein Unterstand (*shelter*) vorhanden ist). Abbildung 3.6 zeigt die zurückgegebenen Informationen markiert als rote Marker auf der Karte. Der *bbox* Bereich, in dem gesucht werden sollte, wurde in 200m Umgebung zum grünen Marker gewählt.

```
1 http://jxapi.osm.rambler.ru/xapi/api/0.6/node[highway=bus_stop][bbox
  =15.392,47.044,15.396,47.048]
```

Listing 3.4: XAPI Anfrage für Point-of-Interest in einer definierten Box.

3 Ausgewählte Themen

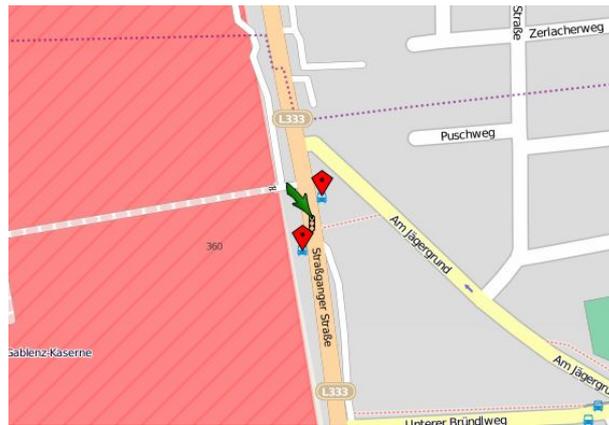


Abbildung 3.6: Rückgabe XML einer XAPI Anfrage mit Darstellung auf einer Karte

3.2.4 Freie Routenplaner auf Basis der OpenStreetMap Karten

Aus den gut dokumentierten OpenStreetMap Daten haben sich in den letzten Jahren einige Routenplaner als Alternative zu jenen von Google Maps oder Bing Maps entwickelt. Wie in Abbildung 3.5 zu erkennen ist, haben die OSM Routenplaner den Vorteil, auf besser dokumentierte Wege zugreifen zu können. Dies gilt vor allem für Fußwege, Wanderwege oder Radfahrwege. Eine Übersichtswebsite⁵ bietet eine Auflistung aktueller Routenplaner für OSM Daten sowie deren Funktionsumfang (Routing für Auto/Fahrrad/Fußgänger, OpenSource, API Zugang etc.). In weiterer Folge werden nun drei ausgewählte Routenplaner, die für die Umsetzung dieses Projektes interessant sind, näher beschrieben. Diese sind das OpenRouteService, YOURS, und der Planer von Cloudmade.

OpenRouteService

Das OpenRouteService (ORS)⁶ sticht durch seinen Funktionsumfang von den anderen hervor, jedoch ist es im Moment (Stand: 14.04.2013) nur für Europa verfügbar. Dessen Funktionalität kann in folgende Punkte unterteilt werden:

- *Route Service*: Der Routenplaner benötigt zu Beginn einen Start- und Endpunkt der gewünschten Strecke. Optional können zur besseren Beschreibung weitere Wegpunkte hinzugefügt werden. Vor der Berechnung muss angegeben werden, ob die Route für »Auto«, »Fußgänger« oder »Radfahrer« optimiert werden soll. Wird Radfahrer ausgewählt, ist es zusätzlich noch möglich, zwischen »Mountainbike«, »Rennrad«, »sicherster Weg« und »bevorzugte Fahrradwege« zu wählen. Hier ist zu sehen, welche genauen Streckenberechnungen durch die exakten Daten der OSM Karten möglich sind. Wie bei Neis und Zipf (2008) beschrieben, hält sich der Routenplaner an die vom OGC definierten OpenLS Standards.

⁵http://wiki.openstreetmap.org/wiki/Routing/OnlineRouters#Route_service_comparison_matrix – zuletzt abgerufen am 03.04.2013.

⁶<http://openrouteservice.org/> – zuletzt abgerufen am 03.04.2013.

3 Ausgewählte Themen

- *Directory Service*: Dieses Service bietet die Möglichkeit, nach POI's zu suchen. Dazu kann angegeben werden, ob POI's in einem gewählten Umkreis eines gesetzten Punktes liegen sollen, ob POI's entlang einer Route angezeigt werden sollen, oder ob der nächstgelegene POI angezeigt werden soll. Durch Angabe des Map Features kann nach der gewünschten Einrichtung gesucht werden (z. B. Kaffee, Kino, Parkplatz, etc.).
- *Location Utility Service*: Bietet die Möglichkeit durch Geocodierung und Reverse Geocodieren eine Adresse in eine geografische Position umzuwandeln et vice versa.
- *Accessibility Analysis Service*: Ein weiteres sehr interessantes Service ist diese Erreichbarkeitsanalyse. Durch Setzen eines Punktes auf der Karte und Angabe von Minuten wird ein Polygon erstellt, das den erreichbaren Bereich markiert. Nicht exakt dokumentiert ist, ob dieses Polygon die Erreichbarkeit für Auto, Fußgänger oder Fahrradfahrer zeigt. Eine Interpretation lässt aber schließen, dass es sich um die eines Kraftfahrzeug handelt.
- *Emergency Route Service*: Dieser Service bietet die Möglichkeit, gefährliche Bereiche durch Polygone zu definieren. Die Route wird dann so berechnet, dass diese Bereiche gemieden werden.

YOURS

Yet another OpenStreetMap Route Service (YOURS) ist ein weiterer Routenplaner auf OSM Basis. Die Demonstrationsseite zeigt, dass das Erstellen der Route mittels Setzen von Punkten und Hinzufügen von Wegpunkten ähnlich abläuft wie jenes beim ORS. Die Arten der Transportmittel, für welche die Route optimiert werden kann, sind dieselben wie beim OpenRouteService, erweitert durch die Punkte »Moped«, »Mofa« oder »Public Service«. Ein Vorteil dieses Routing Engine ist, dass er OpenSource zur Verfügung steht und lokal auf einem Server installiert werden kann. Des Weiteren bietet YOURS eine API an, die eine Route im XML Format zurückgeben kann. Durch Angabe von Attributen können zusätzliche Informationen z. B. das gewünschte Rückgabeformat oder Routeninstruktionen angefordert werden. YOURS bietet laut Spezifikation eine weltweite Abdeckung.

Cloudmade

Ähnlich YOURS ist auch das Routing von Cloudmade. Als Transportmittel lässt sich hier jedoch lediglich »Auto«, »Fußgänger« oder »Radfahrer« angeben. Es ist ebenso eine API vorhanden, welche allerdings erst nach einer Registrierung bei Cloudmade zur Verfügung steht. Dieses Service ist weder Open Source noch ist es möglich, es lokal zu installieren. Jedoch ist eine Servicegarantie von 99,9% vorhanden und die globale Abdeckung macht es zu einem Kandidaten für die Umsetzung dieses Projektes.

3.2.5 Freie Höhendatendienste

Ein weiterer Dienst, zu welchem Alternativen zu Google Maps und Bing Maps vorhanden sind, ist die Abfrage nach Höhendaten in Bezug auf eine geografische Koordinate. Webseiten, welche eine API zur Abfrage anbieten, wären u. a. MapQuest oder GeoNames. Die *Faire-Use-Policy* beschränkt die Koordinaten pro Anfrage nach Höhendaten, jedoch nicht die Anzahl an Anfragen pro Tag. Bei Fraguada (2013) wurde ein Vergleich zwischen Mapquest, GeoNames

und der Google Elevation API durchgeführt. Das Ergebnis zeigt, dass die beiden freien Dienste Mapquest und GeoNames nicht mit der Abdeckung und Qualität der Google Elevation Daten konkurrieren können. Dies resultiert der Autor aus den unterschiedlichen Daten, die zur Berechnung der Höhe herangezogen werden. Die beiden freien Dienste beziehen ihre Daten aus den Höheninformationen des *SRTM (Shuttle Radar Topography Mission)*. Diese wurden im Zuge einer Weltraummission von der NASA mit einem Shuttle, welches mit Radar die Erdoberfläche vermessen hat, aufgenommen und zur Verfügung gestellt (Gesch, Muller & Farr, 2006). Allerdings weisen diese Daten Fehler auf, wie z. B. Fehlpixel an Stellen von starker Geländeneigung, wo des Radarsignal nicht korrekt oder gar nicht zum Shuttle zurück reflektiert wurde (Ernesto Rodríguez & Belz, 2006). Im Vergleich dazu wirken die Daten von Google Maps viel genauer. Hier werden die Höhendaten von mehreren Providern (mehr als 100 verschiedene) bezogen. Diese Summe aus Daten ermöglicht eine bei weitem bessere Abdeckung als jene bei den SRTM Daten.

3.3 Zusammenfassung

Dieses Kapitel befasste sich mit Themen, die als Grundlagen für die Erstellung des Projektes im Rahmen dieser Masterarbeit gesehen werden können. Es zeigt die Entwicklung von satellitengestützten Positionierungssystemen, die schließlich die Möglichkeiten bereitstellten, geographische Informationen zu sammeln. Der Bedarf an Geodaten kann - wie dieses Kapitel zeigt - auf zwei Arten gedeckt werden. Zum einen mit proprietären Diensten, zum anderen mit freien Diensten. Erstere zeigen zumindest Vorteile bei der Bestimmung von Höhendaten, da sie durch Verwendung mehrerer Quellen exakte Informationen liefern. Nachteile ergeben sich jedoch wegen fehlender Wege- und Straßeninformationen beim Planen von Routen. Durch die Analyse der freien Geodaten-Dienste wurden Alternativen zu Google und Bing aufgezeigt.

Aufbauend auf diesen Erkenntnissen betrachtet das folgende Kapitel bereits vorhandene Anwendungen, die sich mit Geocaching und Outdoor Sport in Kombination mit Routenplanern befassen. Die Analyse dieser Anwendungen zeigt, in welche Richtung das Projekt gehen wird, welches im Zuge dieser Masterarbeit entwickelt wurde. Auf Basis dieser Erkenntnisse wird in Folge der Implementierungsprozess aufgebaut.

4 Aktuelle Anwendungen

Der folgende Abschnitt analysiert und evaluiert Webportale und Smartphone Applikationen (kurz Apps), welche sich mit den Themen Freizeit und Sport in Verbindung mit GPS befassen. Zu Beginn werden klassische Sportportale analysiert, bei denen u. a. Routen analysiert und ausgewertet werden. Des Weiteren werden auch Portale betrachtet, die sich mit Geocaching befassen. Dazu werden Webportale mit „klassischem“ Geocaching analysiert, bei dem es Ziel ist, gewisse Koordinaten zu erreichen und dadurch dortige virtuelle Informationen oder Gegenstände einzusammeln. Neben der Möglichkeit, Geocaching auf Punkte anzuwenden, besteht die zusätzliche Alternative, Strecken zu Geocachingobjekten zu machen. Daher wird diese Analyse auch eine Website betrachten, bei der virtuelle Rennstrecken gestaltet werden können.

Die immer kostengünstiger werdende GPS Technik sowie die zusätzliche Möglichkeit der exakten Positionsbestimmung mit Smartphones hat auch den Weg zu Sport- und Freizeitaktivitäten gefunden. War es früher nur professionellen Sportlern vorbehalten, durch teure GPS Aufnahmegeräte exakte Informationen über Distanz und Höhenmeter zu bekommen, so sind diese Geräte in den letzten Jahren einer breiteren Masse verfügbar gemacht worden. Dies führt dazu, dass GPS-Empfänger in den verschiedensten Bereichen zum Einsatz kommen. Neben ihrer eigentlichen Aufgabe, Personen das Navigieren im Gelände zu erleichtern, werden diese heutzutage auch in vielen anderen Gebieten eingesetzt. Z. B. sind Schuhe erhältlich, in denen bereits ein GPS-Empfänger eingebaut ist. Diese Schuhe der Firma Aetrex¹ sind für den medizinischen Bereich entwickelt worden. Sie sollen dem Pflegepersonal von demenz- oder alzheimerkranken Patienten dabei helfen, mit Hilfe der Ortungsfunktion im Schuh die Position der Patienten ausfindig zu machen, da es immer wieder vorkommt, dass diese Patienten nicht mehr zurückfinden (Netzwelt.de, 2013). Ein alternativer Einsatzbereich, in dem die GPS Technik Einzug gefunden hat, ist die Positionsbestimmung von Haustieren. So sind GPS Empfänger erhältlich, die z. B. einer Katze auf das Halsband montiert werden können. Über ein Webtool oder ein App kann der Besitzer von zu Hause aus die aktuelle Position des Tieres abfragen (helpster.de, 2013).

Für diese Arbeit wird der Einsatz von GPS im Sport- und Freizeitbereich genauer analysiert. Repräsentativ für den sportlichen Bereich wird *Runtastic*², *Endomondo*³ und *Strava*⁴ betrachtet. Diese Portale bieten sowohl Smartphone Apps an, welche sportliche Leistung aufzeichnen, als auch ein Webportal, in dem GPS Daten hochgeladen werden können. Die sportlichen Aktivitäten, die durch dieses Webportal analysiert werden können, sind u. a. das Laufen, Radfahren, Wandern, Rugby, Kite Surfen oder Tauchen. Eine spezielles Augenmerk wird hier auf die Aufbereitung der Routeninformationen, die Auswahlmöglichkeiten der Karten sowie die Möglichkeiten, Routen mit Hilfe eines Routenplaners zu erstellen, gelegt.

¹<http://www.aetrex.com/aetrex-gps-shoe/?cat> – zuletzt abgerufen am 30.05.2013.

²<http://www.runtastic.com> – zuletzt abgerufen am 26.04.2013.

³<http://www.endomondo.org> – zuletzt abgerufen am 09.04.2013.

⁴<http://www.strava.com/> – zuletzt abgerufen am 26.04.2013.

Neben dem Einsatz bei sportlichen Freizeitaktivitäten findet GPS immer öfter bei »virtuellen Outdoor-Spielen« Verwendung. Neben dem klassischen Geocaching, bei dem es Ziel ist, geographische Koordinaten aufzusuchen um dort versteckte Gegenstände zu finden, entwickeln sich neue Spiele zu großen virtuellen Szenarien, in welchen Objekte und Orte der wirklichen Welt integriert werden. Als Vertreter dieser neuen Art des Geocaching wird das von Google entwickelte Spiel *Ingress*⁵ betrachtet.

4.1 Webportale und Apps zur Analyse von Sport- und Freizeitaktivitäten

Die Recherche ergab eine Vielzahl von verschiedenen Webportalen und Apps, welche zur Analyse von sportlichen Aktivitäten herangezogen werden können. Repräsentativ werden drei Vertreter, die sowohl durch ein App als auch über eine Webseite verwendet werden können, analysiert. *Runtastic* und *Endomundo* bieten diverse Möglichkeiten, Aktivitäten aufzuzeichnen und zu analysieren. Diese zeigen sportliche Verläufe an, woraus der Athlet z. B. sehen kann, wie oft trainiert wurde, oder wie viele Höhenmeter im letzten Monat geschafft wurden. Alternativ zu diesen Funktionalitäten bietet die Webseite *Strava* die Möglichkeit, virtuelle Rennstrecken zu erstellen. Durch ein Ranking wird ersichtlich, welcher User wie lang für den definierten Rennstreckenbereich benötigt hat. In weiterer Folge werden diese Portale einzeln genauer analysiert.

4.1.1 Runtastic

Runtastic bietet eine Plattform für diverse Sportarten (Laufen, Rennradfahren, Ski-Langlauf etc.) zur Analyse und Verbesserung der Leistungsfähigkeit. Eine Aktivität wird entweder durch das Aufnehmen einer Route direkt mit einem Smartphone in Verbindung mit der *Runtastic* App angelegt, oder die Route wird durch den Upload einer GPX oder TCX Datei ermöglicht. Durch Bestätigen des Uploads wird eine Aktivität der gewählten Sportart erstellt und die Route auf der Karte präsentiert.

Präsentation von Geoinformationen

Zur Auswahl an Basiskarten stehen zur Verfügung: Google Maps (Satellit und Grundkarte), OpenStreetMap und OpenCycleMap (siehe Abbildung 4.1). Neben der Präsentation der Route auf der Karte werden Marker erstellt, die nach jedem Kilometer gesetzt werden. Diese können optional entfernt werden. Das Höhen- und Geschwindigkeitsprofil wird in einem separaten Diagramm angezeigt. Durch Platzieren des Maus-Cursor über dem Diagramm werden einerseits Zusatzinformationen angezeigt, wie z. B. genaue Distanz an dieser Stelle oder Pace, andererseits wird auf der Route in der Karte ein Marker gesetzt, welcher der Distanz auf dem Diagramm entspricht. Dieser Vorgang ist nur in eine Richtung vorhanden, d. h. durch Platzieren des Cursor auf der Route wird im Diagramm nicht gezeigt, wo man sich gerade befindet. Zusätzlich werden Werte wie Gesamtdistanz, Dauer und Höhenmeter aufwärts bzw. abwärts angezeigt.

⁵<http://www.ingress.com/> – zuletzt abgerufen am 03.06.2013.

4 Aktuelle Anwendungen

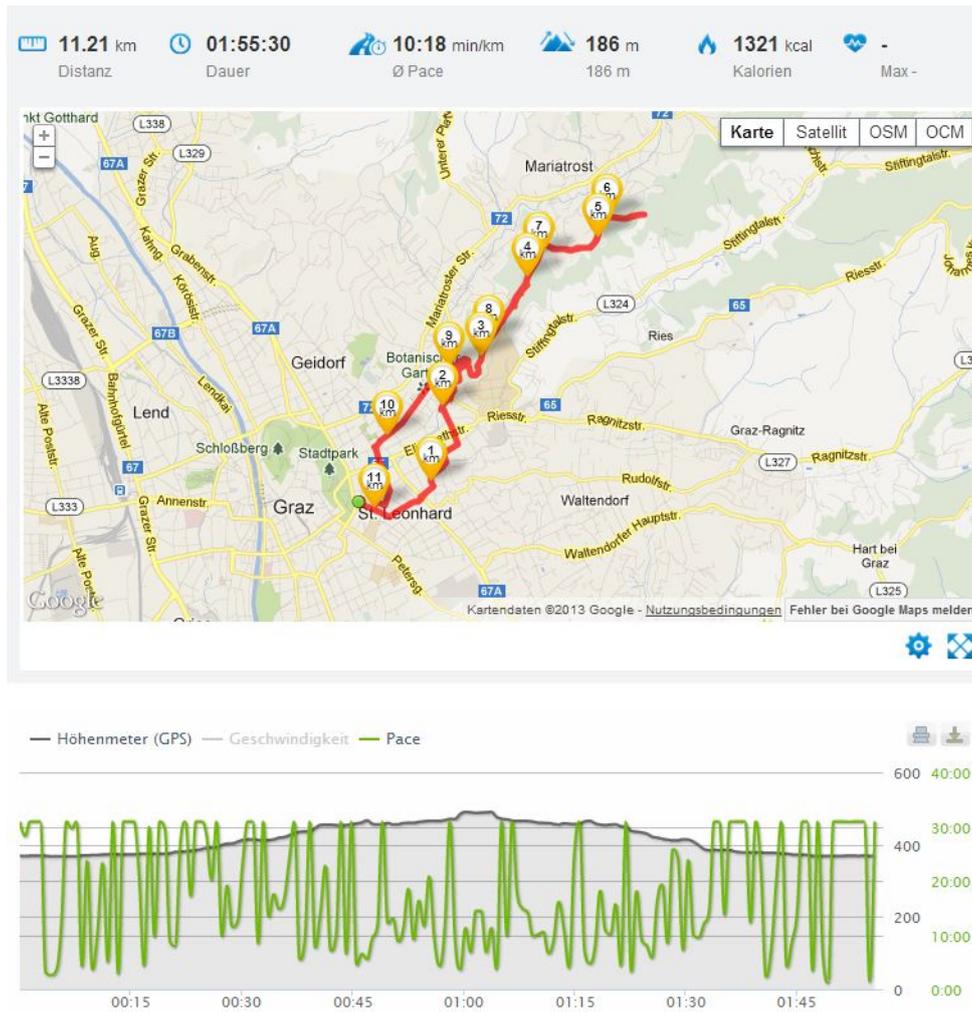


Abbildung 4.1: Präsentation der Strecke in Kombination mit dem Höhenprofil auf *Runtastic*.

Funktionsweise des Routenplaners

Runtastic bietet zusätzlich einen Routenplaner zum Erstellen von Aktivitäten an. Durch Setzen von Wegpunkten auf Straßen oder Wegen werden Routen erstellt. Sollte die Route abseits von Straßen festgelegt werden, ist es möglich, die „Ausrichtung an Straßen“ zu deaktivieren. Danach werden die einzelnen Punkte mit normalen Geraden verbunden. Die angebotenen Straßenkarten sind nur jene von Google Maps, wo sehr viele kleine Wald- und Wanderwege nicht ausgewiesen werden, was eine exakte Routenplanung schwierig gestalten lässt. Des Weiteren besteht die Möglichkeit, eine bereits upgeladete Route im Routenplaner zu erweitern. Jedoch ist eine Verschiebung vorhandener Wegpunkte nicht möglich, wodurch ein nachträgliches Editieren der Route ausgeschlossen ist.

Motivatoren

Die Motivatoren, sportliche Aktivitäten mit dieser Webseite aufzuzeichnen, liegen zunächst an übersichtlich und gut strukturierten Trainingsverläufen. Tabellen und Diagramme zeigen, wie der Trainingsverlauf der letzten Wochen ausgesehen hat, wie viele Kilometer gelaufen wurden oder auch wie lange trainiert wurde. In der Community von *Runtastic* ist es weiters möglich, live ein Training zu verfolgen, sollte das Training mit dem App aufgezeichnet werden. Während der Aktivität können über das App Motivatoren, wie Applaus oder Jubel, an den Athleten gesendet werden.

Gesamteindruck

Die Webseite ist sehr übersichtlich gestaltet und bietet gute Trainingsübersichten in tabellarischer Form. Die Präsentation von Routen und deren Informationen sind ebenfalls zufriedenstellend. Der angebotene Routenplaner ist jedoch von seiner Bedienbarkeit und der Tatsache, dass nur Google Maps Karten ausgewählt werden können, kaum zu verwenden. Zusätzlich ist die Tatsache, dass bestehende Routen durch Verschieben von Wegpunkten nicht verändert werden können, als negativ zu betrachten.

4.1.2 Endomondo

Grob lässt sich die Seite *Endomondo* mit dem Funktionsumfang von *Runtastic* vergleichen. Zusätzliche Features wären z. B die Möglichkeit, Wettbewerbe zu erstellen. Beim Erstellen einer Aktivität hat man die Möglichkeiten, eine Route aus einer Datei zu importieren, bestehende Routen zu verwenden, eine neue Route mit Hilfe eines Routen-Planers zu erstellen sowie keine Route auszuwählen. Bei Letzterem ist es möglich, Werte wie z. B. wie viele Kilometer und wie lange dafür gebraucht wurde, selbst einzutragen. Bei der Auswahl an bestehenden Routen ist der Benutzer nicht nur auf seine eigenen Routen beschränkt, viel mehr kann auf alle bestehenden Routen im System zugegriffen werden.

Präsentation von Geoinformationen

Wie in Abbildung 4.2 zu sehen ist, beschränken sich die Karten bei *Endomondo* auf Google Maps Standard und Satellit. Die Darstellung und Verbindung mit dem Höhen- und Geschwindigkeitsprofil sind identisch mit jenen von *Runtastic*. Als unterschiedliches Features ist es bei *Endomondo* möglich, durch Auswahl der Marker, welche nach jedem Kilometer platziert wurden, Zusatzinformationen wie Pace und Zeit zu erhalten.

Funktionsweise des Routenplaners

Wiederum ist der Routenplaner durch die alleinige Verwendung der Google Maps Karten speziell bei kleineren Wegen sehr beschränkt. Das Zeichnen der Route erfolgt durch Setzen von Wegpunkten entlang der Route. Zusätzlich zum *Runtastic* Routenplaner lassen sich auch Wegpunkte auf der bereits erstellten Strecke verschieben, die dann wieder neu berechnet werden.

4 Aktuelle Anwendungen

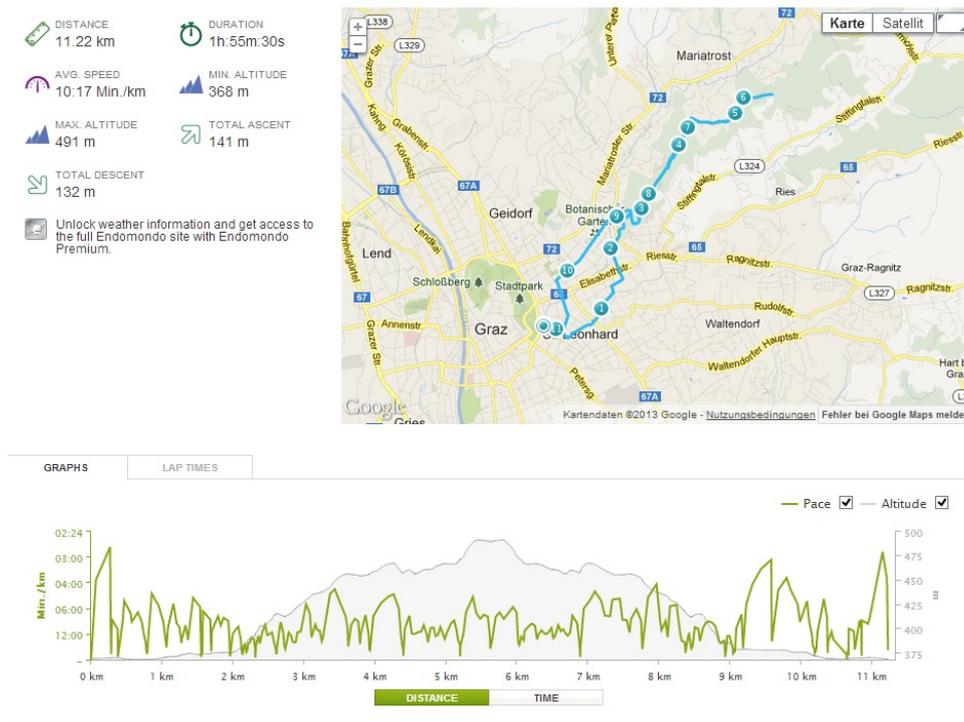


Abbildung 4.2: Präsentation der Strecke in Kombination mit dem Höhenprofil auf *Endomondo*

Ein Verändern bestehender Routen mit Hilfe des Routen-Planers ist nicht möglich. Zusätzliche Tools wären das Hinzufügen von Fotos oder Videos auf der Route.

Motivatoren

Auch *Endomondo* setzt auf gute und übersichtliche Diagramme, welche den Trainingsverlauf präsentieren. Als weitere Motivatoren bietet die Seite die Möglichkeit, Wettkämpfe selbst zu erstellen oder an Wettkämpfen teilzunehmen. Diese Wettkämpfe werden durch einen zu erreichenden Wert in einer gewissen Zeitspanne definiert, z. B. „welcher Athlet schafft mehr Kilometer durch Laufen in den nächsten vier Wochen“. Diese Wettkämpfe sind im Vergleich zu *Runtastic* ein anderer Ansatz, um die Benutzer zum Sport zu motivieren.

Gesamteindruck

Endomondo ist in vielen Punkten mit *Runtastic* zu vergleichen. Beide werden die Bedürfnisse des Sportler befriedigen, auch wenn die Präsentation der Routen und die Möglichkeiten diese zu ändern verbesserbar wären. Der Routenplaner von *Endomondo* scheint in der Bedienbarkeit besser zu sein, jedoch ist das Nichtverwenden von OpenStreetMap Karten ein großer Nachteil, sollte man den Routenplaner als Alternative zum GPS Aufzeichnen der Strecke sehen.

4.1.3 Strava

Die Webseite *Strava* kann im Grunde mit *Runtastic* und *Endomondo* verglichen werden, da sie genauso Routen präsentiert und analysiert. Trotzdem wird hier versucht, auf eine andere Weise zum Sport zu motivieren. *Strava* setzt auf den Wettkampf unter den Benutzern. Die Seite analysiert Routen, findet Segmente darin und misst die Zeit, die für das jeweilige Segment benötigt wurde. Sollte wieder eine Route über dieses Segment führen, werden die Bereiche mit der selbst erreichten Zeit verglichen und eine Rangliste erstellt. Diese Segmente können kurze Abschnitte sein, z. B. eine Steigung beim Mountainbiken, oder auch eine ganze Strecke. Zusätzlich zum automatischen Extrahieren der Segmente aus einer Route können diese auch von den Benutzern selbst bestimmt werden, indem ein Streckenabschnitt durch Verschieben von Markern ausgewählt wird. Es ist kein Routenplaner vorhanden, um Strecken oder Segmente zu erstellen. Die Präsentation der Route und deren Höhenprofil ist wiederum identisch mit den bereits beschriebenen Webportalen. Ein Wiederverwenden von bereits hochgeladenen Routen bzw. ein Verändern dieser ist u. a. wegen Fehlens eines Routenplaners nicht möglich.

Die Möglichkeit, an virtuellen Rennen teilzunehmen, motiviert die Athleten auf eine andere Art, als es bei *Endomondo* oder *Runtastic* geschieht. Dieser Wettkampf kann auch als eine Art von Geocaching gesehen werden, da die Benutzer aufgefordert werden, gewisse Strecken bzw. deren Abschnitte exakt zu befahren oder zu belaufen. Zusätzlich wird noch die gebrauchte Zeit gemessen, um alle Sportler in ein Ranking setzen zu können. Dies ist ein sehr interessanter Ansatz, da der Wettkampf auf virtuellen Rennstrecken und der anschließende Vergleich der gebrauchten Zeit vor allem ambitionierte Sportler motivieren wird.

4.2 Ausgewählte Tools mit Geocaching Funktionalität

Zusätzlich zu den Sportportalen werden im Zuge der Evaluierungen für diese Masterarbeit auch Geocaching Webportale betrachtet. Diese präsentieren in Verbindung mit GPS Geräten eine virtuelle Schatzsuche. Dabei werden zuerst Koordinaten in einem GPS Empfänger gespeichert. Diese Koordinaten müssen anschließend aufgesucht werden, um dort befindliche reale oder virtuelle Gegenstände aufzunehmen.

Die Recherche ergab eine Vielzahl an Smartphone Applikationen, welche auf der Geocaching-Idee aufgebaut sind. U. a. versuchen diese Apps, den prinzipiellen Ablauf von Geocaching mit einem komplexen Spiel zu verbinden. Repräsentativ wird *Geochaching*, *Opencaching*, *Ingress* und *SummitLynx* betrachtet.

Geocaching und Opengeocaching

Der Ablauf und der Funktionsumfang von klassischen Geocaching-Spielen, wie *Geocaching*⁶ und *Opencaching*⁷, ist im Prinzip identisch. Zunächst kann auf einer Karte nach Objekten in der näheren Umgebung gesucht werden. *Opencaching* verwendet Kartenmaterial von Bing Maps, wobei *Geochaching* auf freie Karten zurückgreift, wie OpenStreetMap. Diese Objekte

⁶<http://www.geocaching.com/> – zuletzt abgerufen am 03.06.2013.

⁷<http://www.opencaching.com/> – zuletzt abgerufen am 03.06.2013.

4 Aktuelle Anwendungen



Abbildung 4.3: Darstellung des *Ingress* Spielablaufs. Links: Darstellung eines Portal-Keys. Dieser wird benötigt, um ein Portal einnehmen zu können. Mitte: Karte mit Portalen in der näheren Umgebung. Rechts: Portalübersicht (Hill, 2013)

und deren Koordinaten können dann an GPS Geräte gesendet werden. Durch deren Hilfe werden die Positionen im Gelände aufgesucht. Wird ein Objekt gefunden, können je nach Schwierigkeitsgrad der zu findenden Koordinaten, Punkte gewonnen werden. Viele Objekte sind auch in Gruppen zusammengefasst, sogenannte „Multicaches“. D. h. durch Erreichen aller zur Gruppe gehörenden Koordinaten kann eine höhere Punktezahl gewonnen werden. Diese Seiten bieten nur Punktobjekte an, die aufgesucht und gewonnen werden können. Es ist z. B. nicht möglich, eine ganze Wanderroute als Objekt zu definieren.

SummitLynx

Die Smartphone Applikation *SummitLynx*⁸ versucht, mit Hilfe von Geocaching Personen zum Wandern zu motivieren. Ziel ist es, Berggipfel zu erreichen und dies mit der Applikation zu speichern. In einem Logbuch werden alle erreichten Gipfel aufgezeichnet. Um die Wanderer zusätzlich zu motivieren, werden virtuelle Wettkämpfe und Medaillen angeboten. Ein wichtiger Aspekt dieser App ist sicherlich auch die Möglichkeit, erreichte Gipfel oder gewonnene Auszeichnungen auf Facebook teilen zu können (GooglePlay, 2013).

Ingress

Das von Google vertriebene *Augmented-Reality-Game Ingress*⁹ verbindet ein komplexes virtuelles Spielkonzept mit Geocaching Elementen. Das Prinzip ist es, dass zwei Gruppen von

⁸<http://www.summitlynx.com/> – zuletzt abgerufen am 31.05.2013.

⁹<http://www.ingress.com/> – zuletzt abgerufen am 03.06.2013.

Spielern gegeneinander agieren und wichtige virtuelle Portale, z. B. Gebäude oder Sehenswürdigkeiten, erobern und beschützen. Nachdem ein Portal erobert wurde, wird der Spieler aufgefordert, zum nächstgelegenen Portal zu gehen, um auch dieses zu erobern. Abbildung 4.3 zeigt drei Screenshots des Spielablaufes. Zu erkennen ist u. a. eine Karte, auf welcher die Portale eingezeichnet sind. Kritiker und Datenschützer sehen hier den eigentlichen Sinn der Applikation: Die Spieler merken nicht, dass sie ständig Daten für Google sammeln. Diese Daten kann Google dann z. B. dafür verwenden, um die Navigation von A nach B ermöglichen, wobei A und B die Position zweier Portale im Spiel sind (Lobo, 2013). Trotz dieser Kritiken erfreut sich die Applikation großer Beliebtheit, mit 500.000 Spielern im Mai 2013 (Dalenberg, 2013).

4.3 Zusammenfassung

Dieses Kapitel zeigt die vielfältigen Einsatzbereiche von GPS-Tools im Sport- und Freizeitbereich. Durch den Verbau der Technik in modernen Smartphones entwickelten sich daraus viele Applikationen, die versuchen, GPS in Anwendungen zu integrieren. Dies bringt Sportlern die Möglichkeit, das Training besser und effizienter zu kontrollieren und auszuwerten. Die Apps bieten Trainingstagebücher an, um sportliche Verbesserungen feststellen zu können. Durch den Einsatz von Geoching wird u. a. auch versucht, Spieler für Outdoor-Aktivitäten zu motivieren. So wird es möglich, mit dem Smartphone Gipfelsiege zu speichern, und diese in einer Social-Community zu teilen. Wie Spieler jedoch auch ohne ihr Wissen zu Datensammlern für große Firmen werden, zeigt das Beispiel von *Ingress*.

In dem nun folgenden Kapitel wird eine Möglichkeit diskutiert und präsentiert, die den sportlichen Aspekt des Austausches von Leistungen in einer Social Media Umgebung auf eine innovative Art und Weise realisiert. Dazu werden zu Beginn die in diesem Kapitel analysierten Anwendungen auf ihre Vor- und Nachteile untersucht. Daraus leitet sich das im Zuge dieser Masterarbeit entwickelte Webportal ab. Durch die Kombination und den Einsatz verschiedener GPS- und Geocaching Techniken motiviert dieses Portal die Sportler auf eine neue Art und Weise.

5 Anforderungen und Design

Nachdem im vorigen Kapitel auf die aktuellen Anwendungen eingegangen und deren Fähigkeiten und Funktionen aufgezeigt wurden, soll nun eine neue Webanwendung präsentiert werden, die versucht, bereits existierende Ideen zu verbessern und erweiterte Funktionalität zu präsentieren. Dieses Kapitel gibt einen Überblick darüber, welche Anforderungen erfüllt werden müssen, um eine Webseite zu erstellen, die den sportlichen, sozialen und geocaching Aspekt vereint.

Zu Beginn müssen die im letzten Kapitel diskutierten Anwendungen auf ihre Vor- und Nachteile untersucht werden, um daraus das neue Webportal definieren zu können. Des Weiteren werden sich funktionale Anforderungen ableiten lassen, die letztlich die grundlegenden Funktionalitäten der Verarbeitung und Präsentation von Geoinformationen behandeln. Hauptpunkte sind, welche Geoinformationen verwendet, wie diese erstellt, welche Karten zur Darstellung präsentiert und wie sportliche Aktivitäten mit Geocaching in Verbindung gebracht werden können. In weiterer Folge gibt die konzeptionelle Architektur einen ersten Überblick über die notwendigen Komponenten sowie über verschiedene Abläufe im System.

5.1 Analyse allgemeiner Anforderungen

Im Zuge dieser Masterarbeit soll die Verarbeitung und Aufbereitung von Geoinformationen auf einer Social Media Webseite umgesetzt werden. Themen, die bereits in den Grundlagen beschrieben wurden, werden berücksichtigt und fließen in die Implementation ein. Im Folgenden werden die im vorherigen Kapitel analysierten Anwendungen speziell auf ihre Vor- und Nachteile untersucht, um das neue Webportal, welches im Zuge dieser Arbeit entwickelt wurde, besser definieren zu können.

Beginnend mit den Sport- und Freizeitwebportalen *Runtastic* und *Endomundo* zeigt die Analyse, dass diese sich bei der Präsentation von sportlichen Leistungen nur sehr gering unterscheiden. Gut und übersichtlich präsentieren beide Portale die Strecke einer Aktivität auf der Karte in Verbindung mit einem Höhenprofil. Alle wichtigen Informationen, wie Distanz oder Höhenmeter, werden übersichtlich ausgegeben. Die Webportal-Umgebungen bieten zusätzliche Funktionen wie persönliche Trainingsverläufe der letzten Monate oder auch Übersichtskarten aller Trainingsstrecken innerhalb der Community. Beide Portale bieten zusätzlich auch einen Routenplaner an, der jedoch im Funktionsumfang als sehr beschränkt angesehen werden kann. Dies liegt zum einen daran, dass der Planer auf Karten dargestellt wird, welche nicht die Dichte an Wegen einer OpenStreetMap Karte besitzen. Dadurch wird vor allem das Erstellen von Strecken auf kleinen Waldwegen oder Pfaden unmöglich. Des Weiteren kann der Routenplaner nicht dazu verwendet werden, bereits existierende Routen zu editieren. Lediglich *Runtastic* bietet die Möglichkeit, bestehende Routen am Anfang oder Ende zu erweitern, jedoch ist eine Änderung des Profils der Strecke nicht möglich. Alternativ wurde das

Webportal *Strava* analysiert, welches sich speziell auf das Anbieten virtueller Rennstrecken spezialisiert. Dies bietet einen guten Ansatz, die Leistung sehr ehrgeiziger Sportler auf definierten Strecken vergleichen zu können. Innerhalb eines Rankings wird angezeigt, wer wie lange für den Abschnitt benötigt hat.

Die Analyse der Geocaching-Anwendungen zeigt die unterschiedlichen Einsatzmöglichkeiten dieser Techniken. *SummitSynx* versucht, den Sportler durch das Setzen von Geocaching-Objekten zu motivieren, einen Berg zu erklimmen. Dadurch wird nicht nur der Bergsieg möglich, sondern auch der Gewinn des dort befindlichen Objektes, welches dann auch mit Freunden in einer Social-Community geteilt werden kann. Jedoch ist die Applikation auf Punkte, in diesem Fall Berggipfel, beschränkt. Es können also keine Strecken definiert werden, wie der Gipfel erreicht wurde, oder wie viel Zeit benötigt wurde. Auch die anderen analysierten Applikationen mit Geocaching-Hintergrund zeigen, wie Menschen zur Bewegung motiviert werden können, allein durch die Bestimmung von Punkten, die erreicht werden müssen, um im spielerischen Szenario das nächste Level erreichen zu können.

Aus den Ergebnissen der Analyse aktueller Anforderungen resultiert nun das neue Webportal. Dieses dient der Aufzeichnung, Analyse und Darstellung von Sport- und Freizeitaktivitäten innerhalb einer Social-Community. In dieser Community ist es möglich, dass Benutzer ihre sportlichen Freizeitaktivitäten aufzeichnen und managen können. Erweitert werden diese Funktionalitäten mit Geocaching-Aspekten welche als Motivatoren eingesetzt werden sollen. Die Benutzer können mit den Freizeitaktivitäten virtuelle Objekte aufsammeln. Diese werden sich in Punkt- und Streckenobjekte unterteilen, wobei bei Streckenobjekten optional eine Zeitmessung möglich ist, was sie zu virtuellen Rennstrecken machen wird.

5.2 Funktionale Anforderungen

Die bereits grob beschriebenen Anforderungen werden in Folge genauer auf ihre funktionalen Feinheiten untersucht. Darin werden notwendige Interaktionen mit den Usern beschrieben, um daraus für die Implementierung wichtige Informationen zu erhalten. Untersucht wird u. a. welche Eingabemöglichkeiten ein Routenplaner benötigt oder wie bereits existierende Routen wiederverwendet und verändert werden können.

5.2.1 Präsentation von Routen und Geocaching-Objekten

Hauptthema wird es sein, die Geodaten, wie Strecken oder Geocaching-Objekte, übersichtlich zu präsentieren. Diese Informationen werden sich in Routen aufteilen, welche als Basis der sportlichen Aktivitäten gesehen werden können, Geocaching-Objekte, die in geografische Punkte und Strecken bzw. virtuelle Rennstrecken unterteilt werden. Dargestellt werden sollen diese Daten auf einer Übersichtskarte. Zur Darstellung soll eine Basiskarte eingeblendet werden, die aufgrund des Detailreichtums eine OpenStreetMap Karte sein soll. Zusätzlich soll es möglich sein, die Basiskarte zu wechseln, um Alternativen zur OSM Karte anbieten zu können. Den Nutzern wird es ermöglicht, durch einfache Bedienung geografische Informationen ein- und auszublenden. Die sportlichen Aktivitäten werden durch die Angabe ihrer Sportart in Kategorien gebündelt und durch entsprechende Marker auf der Karte präsentiert. Durch die Wahl eines Markers soll gezeigt werden, welche Aktivität an dieser Stelle ihren Ausgangspunkt hat.

Des Weiteren soll die Route, die zur ausgewählten Aktivität gehört, in der Karte angezeigt werden.

Zusätzlich zu den Aktivitäten sollen auch die Geocaching-Objekte und die virtuellen Rennstrecken auf der Übersichtskarte präsentiert werden. Handelt es sich um Streckenobjekte, werden auch diese durch Auswählen des Markers dargestellt. Zusätzlich werden Informationen präsentiert (z. B. von wem diese Objekte erstellt wurden), um eine soziale Interaktion zu ermöglichen.

5.2.2 Berechnung der Routendaten

Eine Grundanforderung ist es, eine sportliche Aktivität zu erstellen. Diese Aktivität wird neben einem Namen und einem Datum geografische Informationen über den Verlauf der Route brauchen. Diese Informationen sollen entweder durch den Upload einer GPX-Datei, durch direkte Erstellung mit einem Routenplaner oder durch die Auswahl einer bereits im System vorhandenen Route geschehen. Sollte eine GPX-Datei hoch geladen worden sein, muss diese in einer Karte in Verbindung eines Höhenprofils dargestellt werden. Es soll zusätzlich möglich sein, diese Strecke zu editieren. Durch Setzen von Markern entlang der Route, die zu verschieben oder zu löschen sind, kann eine Veränderung vorgenommen werden. Durch diese Änderungsmöglichkeit wird auch auf den Datenschutz Rücksicht genommen. Beginnt oder endet eine Aktivität direkt »vor der Haustür«, so ermöglicht eine Änderung der Route das Kaschieren des eigenen Wohnorts.

5.2.3 Routenplaner und Höhenmeter

Alternativ zu den GPS Daten aus einem mitgeführten Aufnahmegerät soll die Möglichkeit bestehen, eine Route mittels Routenplaner zu erstellen. Die Anwendung eines Routenplaners wird in drei Fällen Gebrauch finden: 1. Beim Erstellen einer Route von Grund auf, 2. bei der Änderung einer eben erst hoch geladenen GPX-Datei und 3. bei der Änderung einer bereits im System vorhandenen Route. Die Funktionalität des Routenplaners soll den Nutzern so einfach wie möglich gemacht werden. Durch das Setzen von Markern auf der Karte wird zwischen diesen eine Route berechnet. Diese Marker sollen verschoben werden können bzw. soll es möglich sein, sie auch wieder zu löschen. Durch eine Nummerierung soll mehr Übersicht geschaffen werden. Zusätzlich soll zwischen bestehenden Markern neue Marker hinzugefügt werden können, um eine genauere Bestimmung eines Streckenabschnitts zu ermöglichen. Die Basiskarte, auf welcher der Routenplaner bearbeitet wird, soll u. a. wegen des Vorhandenseins von kleinen Wegen im Gelände eine auf OpenStreetMap basierende Karte sein. In Verbindung mit der neu erstellten Route wird in einem Diagramm das Höhenprofil präsentiert.

5.2.4 Wiederverwendung und Änderung von Routen

Alternativ wird die Möglichkeit bestehen, eine neue Aktivität zu erstellen, welche eine bereits vorhandene Strecke im System als Basis hat. Entspricht diese Route nur teilweise der trainierten Strecke, soll es möglich sein, diese Strecke zu verändern. Durch das Setzen, Verschieben oder Löschen von Markern kommt auch hier der Routenplaner zum Einsatz. Anwendung soll

diese Funktionalität vor allem dann finden, wenn z. B. bei wöchentlichen kleinen Läufen oft derselbe Weg Richtung Gelände gewählt wird, und sich erst dort der Streckenverlauf ändert. Durch diese Implementierung kann eine leicht abweichende Strecke von einer bereits existierenden ohne viel Aufwand im System angelegt werden.

5.2.5 Virtuelle Rennstrecken

Auf dem Webportal wird es möglich sein, virtuelle Rennen zu veranstalten. Die Sportart und die Rennstrecke soll selbst bestimmt und erstellt werden können. Bei der Erstellung der Strecke wird wieder auf den Routenplaner und dessen Funktionalität zurückgegriffen. Zur Teilnahme an einem Rennen muss sich der Benutzer an diesem anmelden. Durch diese Teilnahme wird jede Aktivität, bei der die Route mit der Rennstrecke übereinstimmt, mit der benötigten Zeit in die Rangliste aufgenommen.

5.2.6 Geocaching von Punkten und Strecken

Ein alternativer Ansatz um die Benutzer zum Sport zu motivieren wird die Möglichkeit darstellen, Geocaching-Objekte einsammeln und ausstreuen zu können. Diese Objekte werden sich nicht nur auf geografische Punkte beschränken, es werden auch Strecken „sammelbar“ sein. Beim Erstellen wird der Benutzer die Möglichkeit haben, entweder ein Punkt-Objekt, oder mit Hilfe des Routenplaners ein Strecken-Objekt zu erstellen. Erstellt der Benutzer eine Aktivität mit einer Strecke, so wird ein Punkt-Objekt dann eingesammelt, wenn die Entfernung zwischen Objekt und irgendeinem Punkt der Strecke unterhalb einer definierten Toleranzgrenze liegt. Bei Strecken-Objekten muss das gesamte Strecken-Objekt auf der Strecke der Aktivität liegen, um gewonnen zu werden (wiederum innerhalb einer Toleranzgrenze).

5.2.7 Zusätzliche Hilfen

Unter anderem soll im System Geocoding und Reverse Geocoding zum Einsatz kommen. Geocoding wird verwendet, um das Navigieren der Karte zu einem Land, Ort oder einer Straße zu ermöglichen. Durch die Eingabe des Ortsnamens soll die Karte zu diesem Ort zoomen. Dadurch kann z. B. ein Ausflugsziel nach bestehenden Aktivitäten oder Geocaching-Objekten erkundet werden. Reverse Geocoding wird verwendet, um automatisch Positionsangaben an die Aktivität anzuheften. Durch das Einholen von Informationen zu Koordinaten der Route wird u. a. der Ort bestimmt werden können, wo die Strecke erstellt wurde. Diese zusätzlichen Informationen sollen die Suche nach Aktivitäten verbessern.

5.3 Nichtfunktionale Anforderungen

In weiterer Folge soll neben den funktionalen Anforderungen auch auf die nichtfunktionalen Aspekte eingegangen werden. Es wird hier auf Designaspekte Wert gelegt, die speziell bei der Bedienbarkeit und Benutzbarkeit des Routenplaners eine große Rolle spielen werden. Weiters wird die Leistungs- und Effizienzoptimierung angesprochen. Der dritte zu analysierende Punkt

befasst sich mit der Korrektheit der sportlichen Auswertung, in Hinblick auf die Ergebnisse des Routings sowie bei der Berechnung der Strecken und Höhendaten.

5.3.1 Aussehen und Usability

Der Vorgang des Erstellens einer Route wird einen wichtigen Punkt darstellen. Es gilt, eine komplexe Funktionalität so einfach wie möglich aufzubereiten. Den Nutzern muss klar ersichtlich sein, wann sie sich im Modus des Streckeneditierens befinden, und wann nicht. Es muss möglich sein, zu erkennen, wie neue Punkte am Ende der Strecke bzw. zusätzliche Punkte auf der Strecke hinzugefügt werden können. Die Routenplanung muss zwischenzeitlich deaktiviert werden können, um auf der Karte zu navigieren, ohne bereits erstellte Punkte zu verändern. Die Präsentation aller geografischen Informationen auf der Webseite muss auf eine effiziente und übersichtliche Art geschehen.

5.3.2 Leistung und Effizienz

Die Dauer der Routenberechnung und Präsentation darf einen angemessenen Rahmen nicht überschreiten. Im Zuge des Routenplanens sollen immer nur kleine Datenpakete angefragt werden, um effizientes Arbeiten zu ermöglichen. Der Upload von externen GPS Daten über eine GPX-Datei kann sehr viele Informationen mit sich bringen. Diese Informationen müssen durch Optimierung und Komprimierung auf eine Größe gebracht werden, die ein rasches Arbeiten ermöglicht, um nicht unnötig Ressourcen zu verschwenden.

5.3.3 Erweiterbarkeit

Es muss möglich sein, zukünftig zusätzliche Sportarten oder Geocaching-Objekte in dem Webportal hinzufügen zu können. Dies sollte so einfach wie möglich, ohne oder nur mit sehr wenig Implementierungsaufwand, umgesetzt werden können. Dadurch kann das Funktionsangebot der Webseite erweitert werden, um den Benutzern immer neue Motivatoren bieten zu können. Bei der technischen Erweiterbarkeit muss vor allem darauf geachtet werden, dass in effizienter Zeit andere Datenformate, welche GPS Daten halten, der Implementierung hinzugefügt werden können. Des Weiteren muss eine Umgebung für die Darstellung und Analyse der Geodaten gewählt werden, die einen zukünftigen Funktionsausbau zulassen wird.

5.3.4 Flexibilität

Die Verwendung von Standards ist vor allem bei dem Upload bzw. Download von GPS-Daten zu beachten. Hier ist zumindest ein gängiger Standard anzubieten, welcher optional auch mit anderen erweitert werden kann.

5.3.5 Korrektheit

Die Berechnung aller relevanter Sportdaten, wie Distanz, Dauer oder Durchschnittsgeschwindigkeit muss korrekt ablaufen. Ebenso müssen die Routenplaner-Daten und die Höhendaten von Plattformen eingebunden werden, die zuverlässig und korrekt arbeiten. Speziell die Höhendaten dürfen von tatsächlichen Werten nicht abweichen, da die Höhenunterschiede in die Sportdaten einfließen. Sollten nur einzelne Wegpunkte auf der Route falsche Höhendaten erhalten, wäre die gesamte Höhenberechnung und die damit verbundene Leistung des Sportlers verfälscht.

5.3.6 Portierbarkeit und Übertragbarkeit

Speziell nach der Erstellung von Routen könnte der Wunsch aufkommen, diese Informationen in eine Datei zu speichern um sie auch bei anderen Webseiten oder lokal verwenden zu können. Daher sollte eine Schnittstelle bereit gestellt werden, bei der es möglich ist, die erstellte Route als GPX-Datei herunterladen zu können. Auch die Angabe der Zeitdauer soll in der Datei gespeichert werden.

5.3.7 Sicherheitsanforderungen

Nachdem das Webportal eine Social-Community darstellt, muss darauf geachtet werden, alle hinzugefügten Daten mit Privatsphäreneinstellungen zu versehen. Der User muss davon ausgehen können, dass Daten, die er nicht veröffentlichen möchte, auch privat bleiben und von keinem Dritten gesehen werden können. Zusätzlich muss auf die Sicherheit des gesamten Webportals geachtet werden, um zu verhindern, dass durch Angriffe private Daten abgefragt werden können.

5.4 Konzeptionelle Architektur

Nachdem basierend auf aktuellen Anwendungen die Grundzüge für das neue Webportal definiert und die Anforderungen im Groben beschrieben wurden, wird im Zuge der konzeptionellen Architektur eine Vorstellung präsentiert, wie diese Anforderungen umgesetzt werden können. Ausgehend von Abbildung 5.1, die einen Überblick über das Web-System und dessen Komponenten gibt, zeigen die Abbildungen 5.2 und 5.3 die Grundzüge des Ablaufs.

5.4.1 Komponenten im Webportal

Das System kann in zwei Blöcke aufgeteilt werden: Das Webportal und die Datenspeicherung. Die Hauptaufgaben werden vom Webportal übernommen. Diese Aufgaben werden sich in die Social-Community-Komponenten sowie die Geodaten-Komponenten unterteilen.

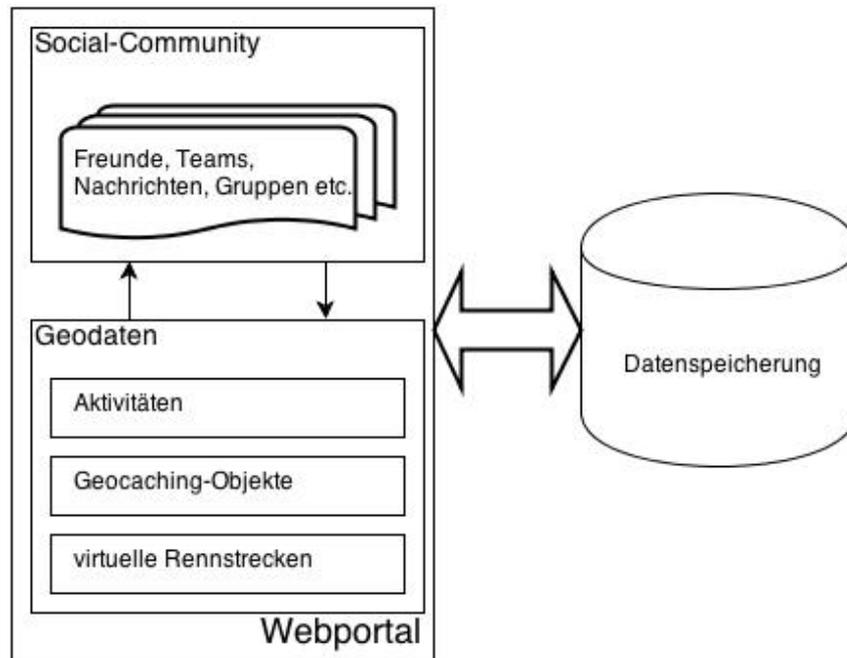


Abbildung 5.1: Darstellung des Systems, welches aus dem Webportal und der Datenspeicherung besteht. Im Webportal befinden sich die Social-Community Komponenten sowie die Geodatenverarbeitung.

Social-Community

Dieser Block ist zuständig für Sozial- und Web 2.0 Funktionalität. Darunter fallen z. B. Themen, wie Freundschaften im System knüpfen, Nachrichten versenden oder Gruppen bilden. Des Weiteren ist er zuständig für das User-Management. Darunter fallen Punkte wie das Anmelden im System, oder statistische Auswertungen von Leistungen des Users.

Geodaten

Die zweite Säule des Webportals wird die Geodatenverarbeitung sein. Darin werden alle Aufgaben übernommen, die sich mit der Präsentation und Bearbeitung von Geodaten befassen. Darunter fallen u. a. das Erstellen, Bearbeiten und Ändern der Routeninformationen von Aktivitäten, von Geocaching-Objekten sowie von virtuellen Rennstrecken. Wie diese Punkte im System abgearbeitet werden, wird in weiterer Folge genauer beschrieben und mit zwei Ablaufdiagrammen dargestellt:

1. Das Erstellen einer Aktivität, welche repräsentativ für das Erstellen von Geocaching-Objekten sowie virtuellen Rennstrecken gesehen werden kann.
2. Das Zusammenarbeiten von Aktivitäten, Geocaching-Objekten sowie virtuellen Rennstrecken im System.

5.4.2 Erstellen einer Aktivität

Am Beginn der Erstellung einer Aktivität muss eine Auswahl getroffen werden, aus welcher Quelle Streckendaten hinzugefügt werden sollen. Sind Daten mit einem GPS Empfänger aufgezeichnet worden, so sollen diese durch Auswahl von „externer Quelle“ hinzugefügt werden. Alternativ wird das Erstellen einer Strecke mit Hilfe eines Streckeneditors angeboten. Zusätzlich sollen bereits erstellte Strecken, welche sich im System befinden, auswählbar sein. Unabhängig von der Quelle werden im nächsten Schritt die Streckendaten präsentiert, analysiert und berechnet. Dabei sollen alle relevanten Daten erhoben werden, wie z. B. Distanz oder Höhenmeter. Sollte die Streckenführung zu diesem Zeitpunkt noch nicht vollständig bzw. korrekt sein, so ist eine Änderung und eine damit verbundene neue Berechnung möglich. Diese Änderungen benötigen ebenso wie der Routenplaner externe Berechnungen der Streckenführung, weshalb externe GIS-Daten angesprochen werden müssen. Abschließend soll die Aktivität gespeichert werden, wodurch sie im System verfügbar wird.

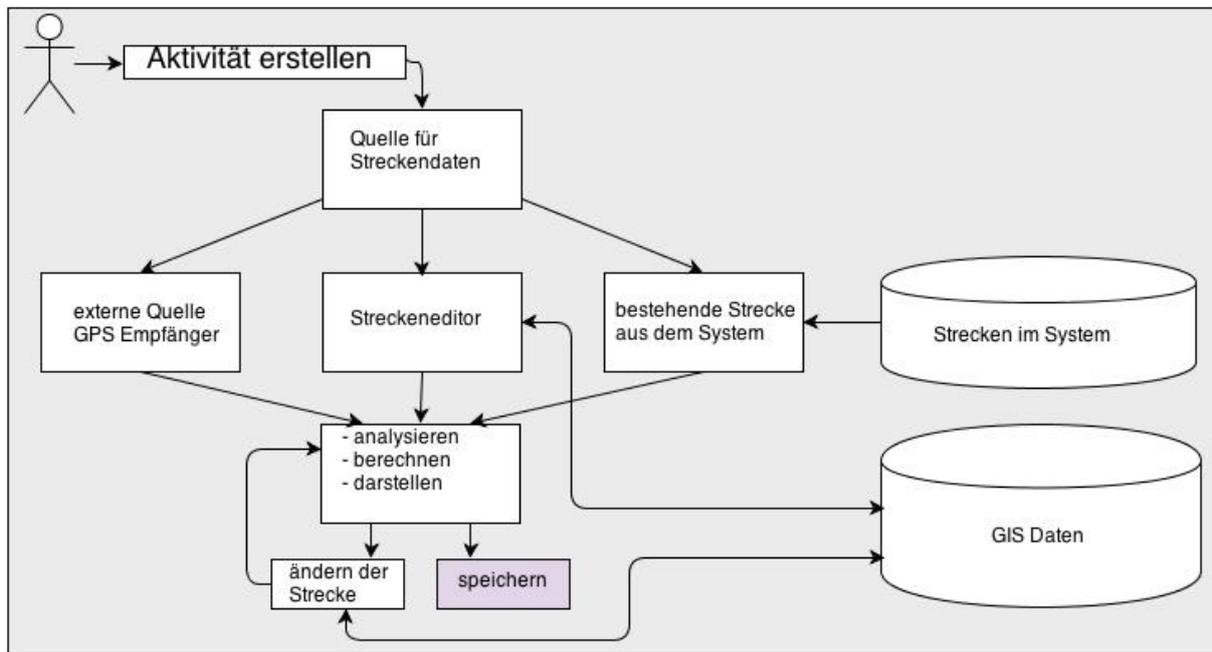


Abbildung 5.2: Ablaufdiagramm des Erstellens einer Aktivität. Dieser Ablauf ist repräsentativ für das Erstellen von Geocaching-Objekten.

Dieser schematische Ablauf entspricht nicht nur dem Anlegen einer sportlichen Aktivität, sondern auf dieselbe Weise werden auch Geocaching-Objekte sowie virtuelle Rennstrecken angelegt. Bei Geocaching-Objekten wird zusätzlich unterschieden, ob es sich um ein Punkt- oder Streckenobjekt handelt, wohingegen die Erstellung der virtuellen Rennstrecke exakt jener der Aktivität entspricht.

Aktivitäten, Geocaching-Objekte und virtuelle Rennstrecken können als einzelne Teile gesehen werden, welche im System zusammenarbeiten müssen. Ausgangspunkt ist die Erstellung einer neuen Aktivität. Sobald diese im System gespeichert wird, muss diese auf mögliche Geocaching-Objekte und Rennstrecken überprüft werden. Dazu müssen zunächst alle Punkt-

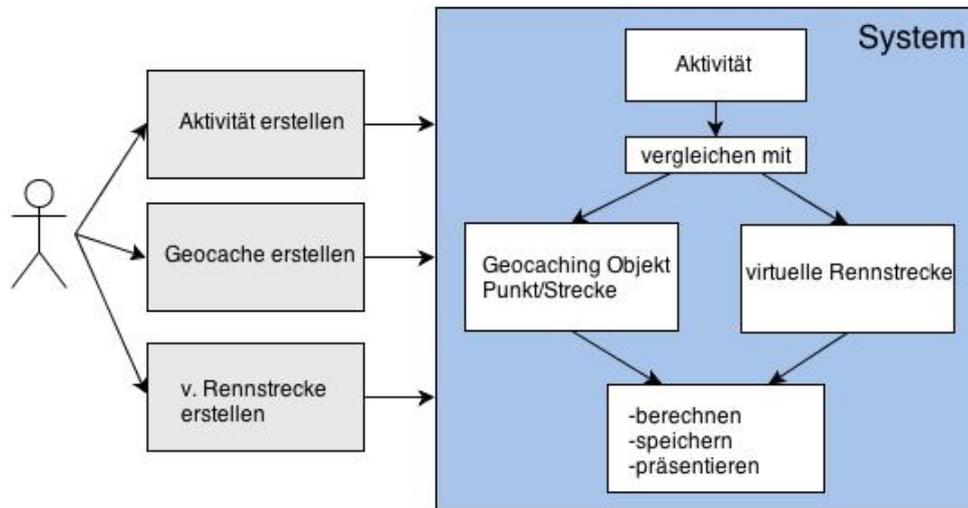


Abbildung 5.3: Zusammenspiel von Aktivitäten und Geocaching-Objekten im System. Nach dem Anlegen oder Ändern einer Aktivität muss kontrolliert werden, ob dazu passende Objekte im System vorhanden sind.

und Streckenobjekte, die mit der Strecke der Aktivität übereinstimmen, bestimmt werden. Bei virtuellen Rennstrecken muss zusätzlich zur Überprüfung der Übereinstimmung die Zeit berechnet werden. Diese Zeit gibt die Dauer an, welche dem Rennstreckensegment der neuen Aktivität entspricht. Nach den jeweiligen Berechnungen müssen etwaige Übereinstimmungen präsentiert und im System gespeichert werden.

5.5 Zusammenfassung

Im Zuge dieser Masterarbeit wird die Verarbeitung von Geoinformationen in einer Social-Media-Webumgebung entwickelt. Dieser Abschnitt befasst sich mit der Analyse der Anforderungen und den Grundzügen des Designs. Basierend auf den Erkenntnissen der Betrachtung aktueller Anwendungen konnten funktionale Anforderungen abgeleitet werden. Daraus resultierte ein Webportal, welches die Funktionalität eines Sport- und Freizeit Webportals mit jenen von Geocaching-Applikationen verbindet. Dafür wird die eigentliche Idee hinter Geocaching, das Aufsuchen gewisser Koordinaten um dort virtuelle Objekte aufnehmen zu können, mit sportlichen Aktivitäten verknüpft. Als weiteren Motivator werden Rennstrecken im System vorhanden sein, bei denen sich die User messen können. Das Zusammenspiel der einzelnen Komponenten im System sowie die notwendigen Schritte der Erstellung einer neuen Aktivität wurde in der konzeptionellen Architektur beschrieben.

Nach der Diskussion der Anforderungen wird im nächsten Kapitel auf die verwendeten Tools und Frameworks eingegangen, welche in der Implementierung zur Umsetzung der Funktionalität eingesetzt werden.

6 Tools und Framework

Bevor mit der Implementierung der in den Anforderungen beschriebenen Komponenten begonnen werden kann, wird auf die verwendete Tools und Frameworks eingegangen. Das Webportal, in welchem die Geodaten verarbeitet werden, wird auf Drupal 7 aufgebaut sein. Dieses Content Management System wird in diesem Kapitel in seinen Grundzügen beschrieben. Die JavaScript Bibliotheken *OpenLayers* sowie *Highcharts* werden für die Bearbeitung und Präsentation von Geodaten auf dem Portal benötigt. Diese Tools unterliegen zunächst einem Entscheidungsprozess, welcher die Gründe deren Verwendung untermauern soll. Im Einzelnen werden die Tools auf Funktionalität, Lizenzen und Verwendung hin untersucht.

6.1 Entscheidungsfindung

Zunächst muss die Entscheidung getroffen werden, mit welchem Framework die WebGIS Umgebung implementiert wird. Die folgenden Punkte fassen die Anforderungen nochmal zusammen:

- Verwendung von OpenStreetMap als Basiskarte, jedoch mit der Möglichkeit, auch Karten anderer Anbieter einzubinden.
- Platzieren, Verschieben und Löschen von Markern bzw. Linien. Diese Objekte werden gebraucht um die Strecken und deren Wegpunkte auf den Karten zu präsentieren.
- Allgemeine Kartenwerkzeuge, die das Navigieren und Zoomen auf der Karte erleichtern.
- Hohe Erweiterbarkeit, um offen und flexibel zu bleiben, falls neben den bisher definierten neue Anforderungen hinzugefügt werden.

Frameworks, die diese Anforderungen erfüllen und für eine Implementierung einer Webanwendung in Frage kommen würden, sind *OpenLayers*¹ oder *Leaflet*². Hier handelt es sich um OpenSource JavaScript Bibliotheken, die für die Arbeit mit Geoinformationen und deren Präsentation in Webanwendungen entwickelt wurden. Die größten Unterschiede zwischen den beiden Bibliotheken liegen in deren Funktionsumfang. Mit *Leaflet* lassen sich die notwendigen Funktionen einfach und schnell umsetzen, jedoch ist im Vergleich zu *OpenLayers* die Erweiterbarkeit und der Funktionsumfang begrenzt.

In vielen Foren und Blogs wird über das optimale Framework zur Umsetzung einer WebGIS Anwendung diskutiert. Hier wird *Leaflet* als neu und innovativ gesehen, jedoch hat es die Nachteile eines geringen Funktionsumfangs. Als optimales Werkzeug sehen die Foren hauptsächlich *OpenLayers*. Nicht nur ob der größeren Auswahl an Funktionen, sondern auch wegen der Vielzahl an Informationen und der großen Community, welche sich mit *OpenLayers*

¹<http://openlayers.org/> – zuletzt abgerufen am 21.06.2013.

²<http://leafletjs.com/> – zuletzt abgerufen am 21.06.2013.

befassen, ist *Leaflet* zu bevorzugen. Diese Punkte waren schlussendlich ausschlaggebend, die *OpenLayers* Bibliothek für die Implementierung zu verwenden.

Da die zu präsentierenden Geoinformationen in eine Social Media Webumgebung eingebettet werden, stellt sich die Frage nach einem Content Management System (CMS) welches in Verbindung mit *OpenLayers* arbeiten kann. Wegen seiner umfangreichen Dokumentation, freien Software und großer Funktionalität ist die Entscheidung auf die Verwendung von Drupal in Version 7 gefallen. Durch die weltweite Drupal Community, die das System verbessert und erweitert, wird die Software als die bevorzugte Entscheidung bei der Entwicklung von großen Social Community Webanwendungen gesehen. Drupal ist eines der meist genutzten CMS überhaupt (Netnode, 2012).

6.2 Drupal 7

Drupal³ ist ein Content Management System basierend auf den Programmiersprachen PHP und MySQL. Es ermöglicht das einfache Verwalten und Organisieren verschiedener Inhalte in einem Webportal. Durch einen modularen Aufbau kann die Standardinstallation der Software schnell auf die gewünschte Funktionalität gebracht werden. Mit der Software lassen sich sowohl kleinere Webseiten als auch umfangreiche Social Community Plattformen qualitativ hochwertig umsetzen. Der Aufbau einer Webseite in Drupal ist zur Gänze ohne Programmierkenntnisse möglich, wodurch die Anwendung einfach und intuitiv wird. Die Nutzoberfläche ist zur Gänze webbasierend, weshalb Drupal in jedem Webbrowser bearbeitet und genutzt werden kann. Zusätzlich lassen sich alle Inhalte und Layouts getrennt voneinander bearbeiten, was die Flexibilität der Software zum Ausdruck bringt (Shreves & Dunwoodie, 2011, S. 3). Durch die große Auswahl an unterschiedlichen Komponenten kann nach dem Baukastenprinzip jede Art von Web-Tool in einer Drupal Website integriert werden. Die Erstellung von Blogs, Wikis, Online-Shops oder Foren ist daher rasch und effizient möglich. Die Software bietet ein Usermanagement, dessen Funktionsumfang modular erweitert werden kann. Dadurch können umfangreiche Nutzerrechte und Benutzerrollen bestimmt werden.

Drupal Lizenz

Das Content Management System Drupal steht unter der *GNU General Public License*. Es handelt sich um eine Open Source Software die frei zur Verwendung steht. Es ist jedem erlaubt, die Software zu verwenden und zu verändern. Die Verwendung der Software ist kostenlos, daher entsteht ein großer finanzieller Vorteil für Unternehmen, sollten sie ihre Webseite mit Hilfe von Drupal betreiben. Durch die Erweiterbarkeit ist eine große Community entstanden, welche die Software wartet und neue Funktionalitäten hinzufügt. Dadurch wird Drupal immer auf dem technologisch neuesten Web-Standard gehalten (Stahl & Schettler, 2012, S. 1).

³<https://drupal.org/> – zuletzt abgerufen am 21.06.2013.

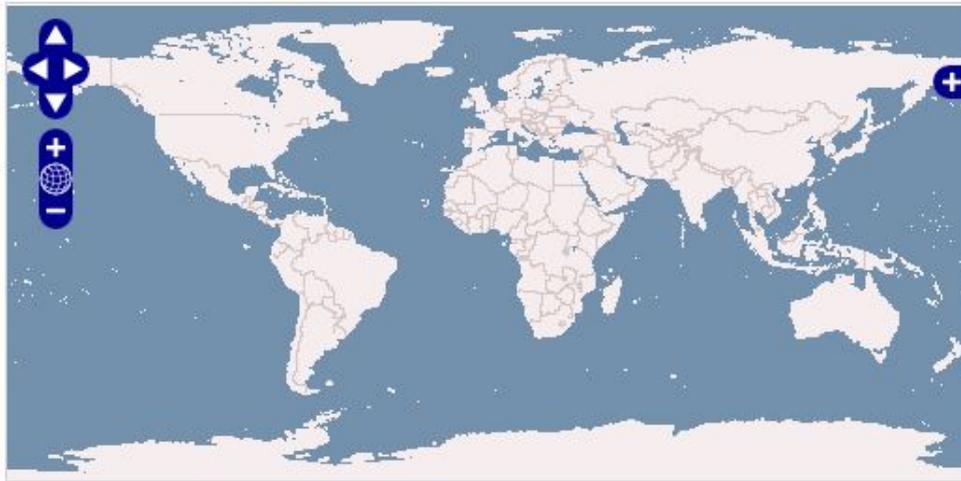


Abbildung 6.1: OpenLayers mit einer eingebundenen Weltkarte und Bedienelementen. Links befinden sich die Navigations- und Zoomelemente, rechts befindet sich ein Layer-Switcher.

Dokumentation und Verwendung

Die offene Software, die oftmalige Verwendung und die gute Dokumentation hat zur Entscheidung beigetragen, Drupal zur Entwicklung zu verwenden. Die freiwillige Entwickler-Community ist mit verschiedenen Kanälen, wie z. B. IRC Chats oder Foren, erreichbar. Zusätzlich werden Drupal Events veranstaltet, bei denen sich Entwickler austauschen können. Drupal wird von vielen großen Unternehmen als Basis der eigenen Webseiten verwendet, u.a sind das Greenpeace, MTV UK oder der NY Observer (Byron, Berry & De Bondt, 2012, S. 1-2). Laut Statistik⁴ beschäftigen sich 761.244 Entwickler aus 228 Ländern mit der Entwicklung und Betreuung von Drupal und dessen Features.

6.3 OpenLayers

Geht es nach den Autoren Jansen und Adams (2010, S. 17), gibt es kein anderes Kartenframework für WebGIS Anwendungen, welches sich mit den Freiheiten und Möglichkeiten von *OpenLayers* vergleichen lässt. Kartendienste unterschiedlichster Anbieter lassen sich mit Hilfe der *OpenLayers* Bibliothek auf einfachste Weise in Webanwendungen integrieren, ohne deren spezielle Einbindungen lange studieren zu müssen. Neben den proprietären Daten lassen sich auch alle freien Geodaten, die nach dem OGC Standard angeboten werden, verwenden. Durch die gute Dokumentation und große Community an Entwicklern existieren viele Beispiele und Tutorials, die das Implementieren einer *OpenLayers* Anwendung erleichtern.

⁴<http://drupalfacts.mogdesign.eu/> – zuletzt abgerufen am 13.05.2013.

6 Tools und Framework

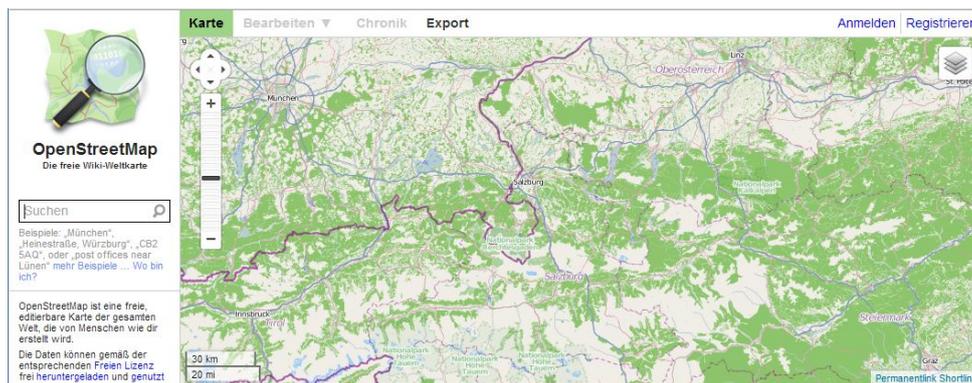


Abbildung 6.2: OpenStreetMap setzt bei der Darstellung der Karte auf OpenLayers.

Funktionalität

OpenLayers ist eine in JavaScript speziell für Web-Mapping Anwendungen entwickelte Umgebung. Es bietet die Grundlagen, um geografische Daten auf einer Kartenoberfläche, welche mit Bedienelementen ausgestattet ist, zu präsentieren. Diese Bedienelemente erlauben u. a. das Navigieren und Zoomen auf den Karten. Diese Werkzeuge stellen mit der Oberfläche die GUI Komponenten der Karte dar, wie in Abbildung 6.1 zu sehen ist. Neben den proprietären Geodaten wie Google Maps oder Bing Maps lassen sich auch die freien Alternativen wie OpenStreetMaps leicht einbinden und präsentieren. Ebenso lassen sich Features, die bereits im Grundlagenkapitel erklärt wurden, wie Web Mapping Services oder Web Feature Services - sollten sie nach den OGC Standards bereitgestellt werden - in *OpenLayers* verwenden.

Ein weiterer großer Vorteil ist die vereinheitlichte API. Dadurch können alle Datendienste unabhängig von der übrigen Programmierung ausgetauscht werden. *OpenLayers* verfolgt eine strikte Trennung von Server eingebundenen Daten und den clientseitigen Interaktionen, welche mit den Daten vorgenommen werden können. Dadurch müssen bei Änderungen von Kartendaten, die über einen Server eingebunden werden, die clientseitigen Interaktionen mit dieser Karte nicht neu implementiert werden (Jansen & Adams, 2010, S. 58).

OpenLayers Lizenz

OpenLayers entstand als freie Alternative auf Basis der Google Maps API im Jahr 2006. Seit diesem Zeitpunkt wurde es immer weiter entwickelt und mit neuen Karten und Funktionalitäten erweitert. Es wird unter einer BSD-Lizenz als freie Software betrieben. Das bedeutet, eine auf *OpenLayers* basierende WebGIS Anwendung darf frei veröffentlicht werden. Die einzige Bedingung, die bei der Veröffentlichung nicht fehlen darf, ist die Copyright-Information. Diese freie Verwendung bezieht sich jedoch nicht auf etwaige eingebundene Daten von Drittanbietern. Natürlich müssen bei Verwendung von proprietären Daten - wie zum Beispiel Google Maps - deren Nutzungsbestimmungen eingehalten werden (Jansen & Adams, 2010, S. 59).

6 Tools und Framework

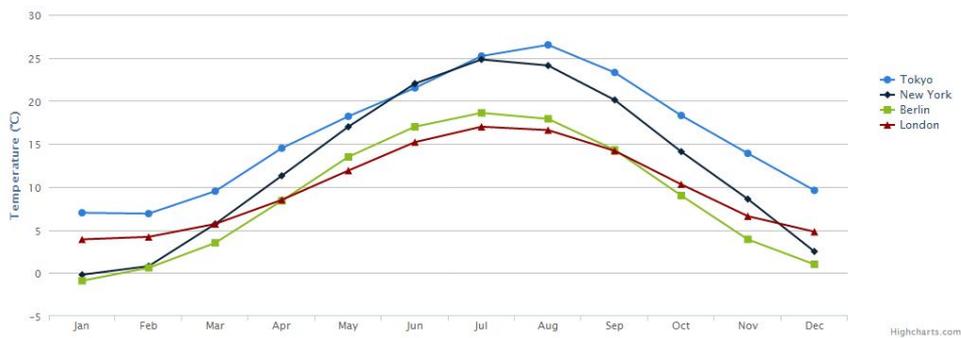


Abbildung 6.3: Darstellung von vier Diagrammen unterschiedlicher Farbe inklusive Legenden und Achsenbeschreibung in Highcharts.

Dokumentation und Verwendung

Auf der Homepage ist eine ausführliche Dokumentation aller Funktionen und Objekte vorhanden. Durch die große Community, welche sich mit *OpenLayers* befasst und darauf basierend Implementierungen durchführt, finden sich im Web zahlreiche Tutorials und Blogs zur Unterstützung. Sollte auch hier keine Hilfe gefunden werden können, wird auf eine Mailingliste⁵ hingewiesen. Bekannte Webseiten, die *OpenLayers* zur Anzeige von geografischen Informationen in Karten verwendet werden sind u. a. *OpenStreetMaps* (Siehe Abbildung 6.2) oder das Weiße Haus⁶.

6.4 Highcharts for JS

Zur Darstellung der Diagramme, speziell bei der Auswertung der Höhendaten, wurde die *Highcharts*⁷ Bibliothek gewählt. Sie wird u. a. auch auf den bereits näher beschriebenen Webanwendungen *Endomundo* und *Runtastic* eingesetzt. Diese Bibliothek bietet eine große Auswahl an verschiedenen Diagrammen, wie Linien oder Tortendiagrammen. Es können mehrere Daten auf einmal visualisiert werden und die graphische Präsentation ist benutzerdefiniert einstellbar. Abbildung 6.3 zeigt ein Liniendiagramm mit vier Datensätzen. Zusätzlich bietet *Highcharts* umfangreiche Funktionen an, wie z. B. Zoomen auf dem Diagramm oder Definieren von Popups mit spezifiziertem Inhalt. Wird die Bibliothek auf einer Non-Profit Webseite verwendet, so unterliegt diese der *Creative Commons Attribution-NonCommercial 3.0 License*, wodurch eine freie Nutzung der Software möglich wird (Highcharts, 2013).

6.5 Zusammenfassung

Dieses Kapitel zeigt Punkte auf, die zur Entscheidungsfindung der verwendeten Tools beigetragen haben. Zur Umsetzung der Geodaten, Verarbeitung und Präsentation, wurde schließ-

⁵<http://drupal.org/mailling-lists> – zuletzt abgerufen am 13.05.2013.

⁶<http://www.whitehouse.gov/change> – zuletzt abgerufen am 13.05.2013.

⁷<http://www.highcharts.com/> – zuletzt abgerufen am 21.06.2013.

lich die umfangreiche *OpenLayers* Bibliothek ausgewählt. Argumente dafür waren vor allem der große Funktionsumfang sowie die Vielzahl an Beispielen und Tutorials, die als Implementationshilfe verwendet werden können. *OpenLayers* bietet alle notwendigen Möglichkeiten, um die Anforderungen, die im speziellen an den Routenplaner der Webanwendung gestellt werden, umsetzen zu können.

Auf Basis der Software Drupal 7 wird das Webportal aufgebaut. Dieses modular aufgebaute Content Management System zeichnet sich vor allem durch seine Vielseitigkeit und Erweiterbarkeit aus. Zusätzlich ist bereits ein *OpenLayers* Modul für Drupal vorhanden, was ein reibungsloses Arbeiten der Bibliothek in der Drupal Umgebung ermöglicht. Zur Präsentation von Höhendigrammen in Kombination mit den Streckeninformationen wurde noch kurz die verwendete *Highcharts* Bibliothek angesprochen. Im folgenden Kapitel wird gezeigt, wie die beschriebenen Komponenten bei der Implementierung eingesetzt worden sind.

7 Implementierung

Dieses Kapitel beschreibt die Architektur und einzelnen Schritte der Implementation, die zur Umsetzung dieses Projekt nötig waren. Zu Beginn wird auf die Grundfunktionalität von Drupal und *OpenLayers* eingegangen, um die darauf aufbauende Implementierung besser verständlich zu machen. Dazu werden die Grundlagen, die Architektur und die wichtigste Terminologie zu Drupal sowie die drei wichtigsten Objekte (*Map*, *Control*, *Layer*) von *OpenLayers* beschrieben. Die Architektur der umgesetzten Webapplikation wird mit einem Dreischichten-Modell beschrieben. Dadurch kann erkannt werden, wie *OpenLayers* mit WebGIS Komponenten zusammenarbeitet und wie diese Funktionalität in das Drupal-System eingebunden wurde. Die Implementierung befasst sich zunächst mit der Umsetzung des Streckeneditors. Es wird gezeigt, welche Objekte und Funktionen von *OpenLayers* verwendet wurden und wie die Darstellung in Verbindung mit dem Höhenprofil durchgeführt wurde.

Um die Anforderungen von Geocaching umsetzen zu können, muss Geodatenverarbeitung in das Drupal-System integriert werden. Dafür muss nach der Speicherung neu erstellter sportlicher Aktivitäten in dem Webportal kontrolliert werden, ob sich in der Nähe der Strecke eine Geocaching-Objekt befindet. Liegt dieses innerhalb einer Toleranzentfernung, so wird das Objekt „eingesammelt“. Abschließend wird in diesem Kapitel auf die Verwendung des *OpenLayers*-Modul für Drupal 7 eingegangen. Dieses wird vor allem dazu genutzt, um alle in der Webapplikation vorhandenen Geoinformationen in einer Übersichtskarte zu präsentieren.

Diese Arbeit ist für die Geodaten-Funktionalität in dem neu erstellten Webportal verantwortlich. Der gesamte Aufbau der Social-Community mit Drupal wird in der Arbeit von Sandrieser (2013) beschrieben.

7.1 Grundlagen zu Drupal 7

In diesem Abschnitt werden Grundlagen zu Drupal und dessen Terminologien beschrieben. Dies sind Voraussetzungen, um die Implementationen, die in Drupal vorgenommen wurden, verstehen zu können. Zunächst wird auf die Terminologie eingegangen, welche die Komponenten des Content Management System beschreiben. Aufbauend auf diese wird die Architektur und Arbeitsweise von Drupal näher erläutert.

7.1.1 Grundlegende Terminologie

Die folgenden Begriffe beziehen sich auf grundlegende Funktionen in Drupal. In der Implementierung wird mit dieser Terminologie gearbeitet, weshalb sie kurz näher erklärt werden:

7 Implementierung

- *Front- und Backend*: In einem CMS wird zumeist mit zwei Ansichten gearbeitet. Als *Frontend* wird die Darstellung des Systems für die Benutzer bezeichnet, und als *Backend* jene, mit welcher der Administrator arbeitet. In Drupal 7 besteht der Hauptunterschied zwischen den beiden Ansichten darin, dass für den Administrator zum einen eine Menüleiste angezeigt wird, zum anderen ist der Inhalt direkt zu bearbeiten (Graf, 2008, S. 32).
- *Node*: Als *Nodes* werden die Inhalte der Webseite bezeichnet. Auf dem Sportportal, welches im Zuge dieser Masterarbeit entwickelt wurde, entspricht z. B. eine Aktivität einer *Node*.
- *Block*: Ein *Block* wird verwendet, um Informationen, Menüs oder Navigationselemente auf einer Drupal Seite zu platzieren. Als Administrator ist es möglich, neue *Blocks* zu erstellen sowie deren Inhalt und Position auf einer Seite zu definieren.
- *Module*: Wie bereits erwähnt, ist es mit Hilfe von *Modulen* möglich, die Funktionalität von Drupal zu erweitern. Unterschieden werden muss zwischen den *Core Modulen*, die für die Standardfunktionalität verwendet werden, und neu hinzugefügten *Modulen*, welche die Funktionalität erweitern.
- *Themes und Templates*: Drupals *Themes* bestimmen das Aussehen der Webseite. Sie bestehen aus CSS und PHP *Templates*. Durch ein *Template* kann eine Ausgabe von Drupal überschrieben werden, um das Design und Aussehen anzupassen.
- *Hooks*: Sie ermöglichen es, einen Request in Drupal abzufangen, zu ändern und zu erweitern. Dadurch wird es möglich, Interaktionen zwischen dem Drupal Core und den Modulen zu definieren. *Hooks* sind in Drupal registrierte Funktionen, die aufgerufen werden, sobald irgendwo im System diese Aktion, welche der *Hook* abfängt, ausgeführt wird. Dadurch können mehrere Module bei einem Request zusammen interagieren (Butcher, Garfield & Wilkins, 2010, S. 14).

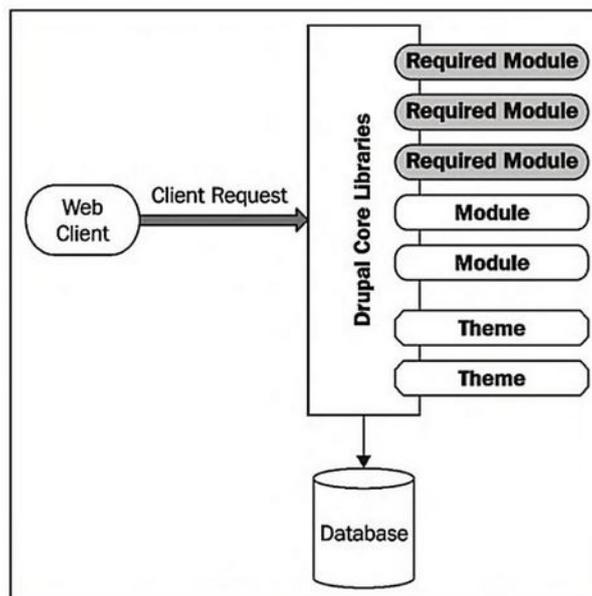


Abbildung 7.1: Schematische Darstellung eines Drupal Requests. Nach einer Anfrage vom Client übernehmen die Core-Module das Datenmanagement zwischen den einzelnen Modulen (Butcher et al., 2010, S. 11).

7.1.2 Architektur und Funktionalität

Abbildung 7.1 zeigt eine schematische Darstellung der Drupal Architektur. Folgend wird der Ablauf eines Requests, welcher diese Architektur durchläuft, beschrieben (Butcher et al., 2010, S. 12):

1. Im ersten Schritt erhält der Webserver eine Anfrage für eine Drupal Seite, z. B. „www.example.org/node/1“.
2. Der Webserver verarbeitet diese Anfrage mit PHP und übergibt die Anfrage „/node/1“ an die index.php.
3. Es startet der Bootstrap Prozess, der entscheidet wie der Pfad „/node/1“ zu verarbeiten ist.
4. Das Node System ladet den Inhalt mit der Identifikationsnummer „1“ aus der Datenbank, oder - sofern vorhanden - aus dem Cache.
5. Durch die Beachtung von Themes und Templates, welche die Präsentation und Gestaltung verändern, wird der angefragte Inhalt im Browser präsentiert.

7.2 Die OpenLayers Grundlagen

Durch Einbindung der *OpenLayers* JavaScript Bibliothek ist es möglich, auf die im *OpenLayers* Container befindlichen Objekte zuzugreifen. Zu diesen Objekten werden Kind-Objekte, Methoden und Attribute angeboten, welche in der Dokumentation¹ eingesehen werden können. In weiterer Folge werden die Objekte *Map*, *Layer* und *Control* näher beschrieben. Sie werden in beinahe allen *OpenLayers* Implementierungen, die im Zuge dieser Masterarbeit umgesetzt wurden, Gebrauch finden. Jede *OpenLayers* Anwendung ist auf einem *Map*-Objekt und auf zumindest einem oder mehreren *Layer*-Objekten aufgebaut. Optional kann die Anwendung mit einem *Control*-Objekt erweitert werden. Daher ist es wichtig, ihre grundlegenden Aufgaben und Fähigkeiten näher zu erläutern.

7.2.1 Map-Objekt

Zu Beginn jeder WebGIS Anwendung - basierend auf *OpenLayers* - steht der Konstruktor einer Instanz eines *Map*-Objekts. Dieses Objekt stellt einen leeren Container dar, in dem alle weiteren Objekte, wie zum Beispiel *Layer*-Objekte, eingefügt werden können. Dieses zentrale Objekt enthält alle Referenzen zu *Layer* und *Control*-Objekten. Der Konstruktor des Objekts wird wie folgt aufgerufen:

```
1 var map = new OpenLayers.Map("div_id", { Optionen });
```

Listing 7.1: Der Konstruktor für das Erstellen eines *Map*-Objekts

Im *div_id* Parameter wird die ID eines HTML `<div>` Objektes angegeben, welches die *OpenLayers* Karte hält. Position, Höhe und Breite dieses *Div* können durch CSS Befehle bestimmt werden. Dies hat auf die *OpenLayers* Implementierung keine Auswirkung. Anstelle von

¹<http://dev.openlayers.org/docs/files/OpenLayers-js.html> – zuletzt abgerufen am 13.05.2013.

Optionen können die standardmäßigen Einstellungen von OpenLayers überschrieben werden. Diese wären z. B. der maximale und minimale Maßstab, Anzahl der Zoomstufen oder eine Beschränkung der zu navigierenden Bereiche.

7.2.2 Layer-Objekt

Mit `OpenLayers.Layer`-Objekte werden alle darzustellenden Elemente, wie Karten, Punkte, Linien oder Flächen visualisiert. Als ein Layer wird z. B. ein eingebundenes WMS bezeichnet, eine Google Maps Karte bzw. Marker oder Pop-ups. Unterschieden wird hier weiters in *Base Layers* und *Non Base Layers*. Als *BaseLayers* werden solche Layer-Objekte bezeichnet, die immer nur allein angezeigt werden können. Es können also keine zwei *BaseLayer* zur selben Zeit in *OpenLayers* aktiviert werden. Vor allem Karten werden als diese Art von Layer definiert, da es wenig Sinn machen würde, über eine Google Maps Karte eine OpenStreetMap Karte zu legen. Alle anderen Layer, also Nicht Base Layer oder auch „Overlays“ genannt, können gemeinsam oder einzeln angezeigt werden. Als Beispiel wird hier eine Instanziierung eines WMS Layers gezeigt.

```

1 var wms = new OpenLayers.Layer.WMS("NASA Global Mosaic",
2   "http://wms.jpl.nasa.gov/wms.cgi",
3   {
4     layers: "modis,global_mosaic"
5   });

```

Listing 7.2: Dieser Codeauszug zeigt den Konstruktor für ein WMS-Layer Objekt.

Der erste Parameter des WMS Layer Konstruktors beschreibt den Namen bzw. die ID des Layers (in diesem Fall „NASA Global Mosaic“). Der Zweite gibt die URL an, von welchem Server die WMS Daten geladen werden. Die letzten Parameter geben zusätzliche Argumente an, die mit der URL übergeben werden.

Anstelle von „WMS“ werden die Namen anderer Layer-Typen gesetzt. Dadurch verändern sich auch die Parameter. Es ist z. B. auch möglich, einen GML oder KML Layer zu erzeugen, mit dem Daten dieses Formats direkt in *OpenLayers* präsentiert werden können. Zu weiteren Typen zählen auch die proprietären Kartendaten von Bing und Google. Sichtbar gemacht werden vorhandene Layer in einem „Layer Switcher“. Damit ist es möglich, bei Vorhandensein mehrerer Layer zu wählen und diese ein- bzw. auszublenden. Wie ein „Layer Switcher“ erstellt wird, wird im folgenden `Control` Abschnitt näher erläutert. Die genauere Verwendung und die Funktionen des Layer-Objektes werden wiederum direkt in der Implementierung beschrieben.

7.2.3 Control-Objekt

`Control`-Objekte ermöglichen es, direkte auf die Karte oder auf einen Layer Einfluss zu nehmen. Auf der Karte ermöglichen diese Objekte grundlegende Funktionalitäten wie Zoomen oder Navigieren. Bei einem Layer werden mit `Control`-Objekten u. a. Events registriert. Diese Events erlauben z. B. das Ändern eines Layers oder das Verschieben eines Markers.

7 Implementierung

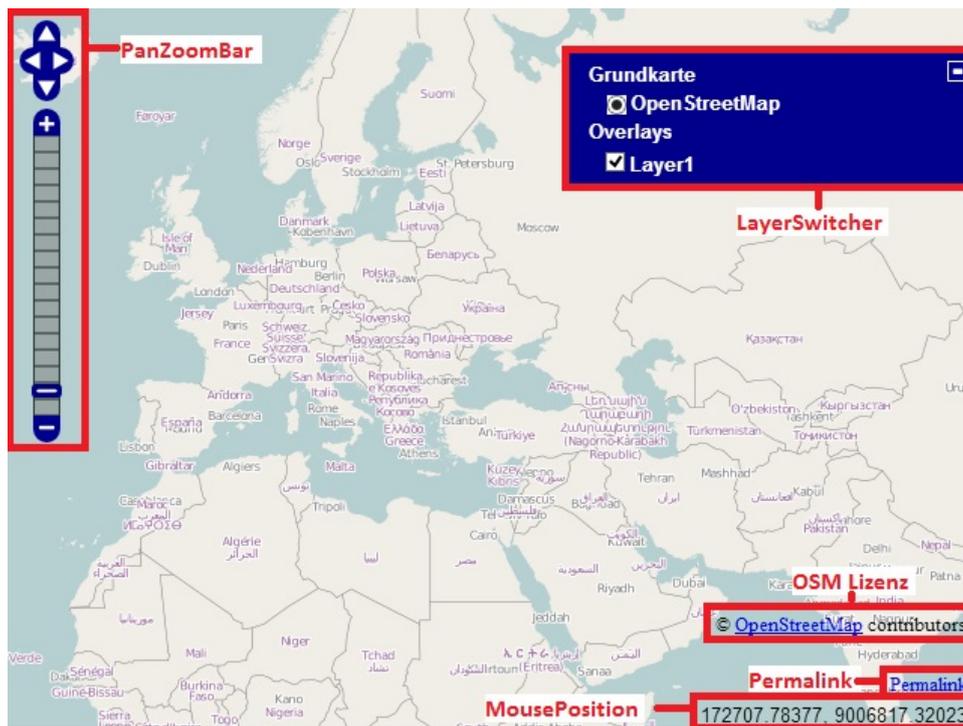


Abbildung 7.2: Darstellung der wichtigsten OpenLayers Control-Objekte auf einer OpenLayers Map.

```
1 map.addControl(new OpenLayers.Control.PanZoomBar());
2 map.addControl(new OpenLayers.Control.MousePosition());
3 map.addControl(new OpenLayers.Control.Navigation());
4 map.addControl(new OpenLayers.Control.Permalink());
5 map.addControl(new OpenLayers.Control.LayerSwitcher());
```

Listing 7.3: Konstruktoren der wichtigsten OpenLayers Control-Objekte.

Listing 7.2 zeigt die Aktivierung der Control-Objekte auf der Map. Daraus resultiert Abbildung 7.2. Im Einzelnen übernehmen die Control-Objekte folgende Aufgaben:

- *PanZoomBar*: Ermöglicht das Navigieren und Zoomen der Karte, bestehend aus vier Richtungspfeilen sowie einen Schieberegler, welcher die Zoomstufe bestimmt.
- *MousePosition*: Gibt die Koordinaten an, an welcher Position sich gerade die Maus befindet.
- *Navigation*: Erlaubt das Navigieren der Karte mit Maus-Events. So kann die Karte z. B. durch Halten der linken Maustaste bewegt werden.
- *Permalink*: Ermöglicht es, die aktuelle Position und Zoomstufe der Karte in die URL zu übertragen.
- *LayerSwitcher*: Ein zentrales Element aller *OpenLayers*-Anwendungen. Falls mehrere Basiskarten oder Overlays vorhanden sind, können diese mit dem *LayerSwitcher* ausgewählt werden. Es kann immer nur eine Basiskarte gewählt werden, jedoch können mehrere Overlays gleichzeitig dargestellt werden.

7 Implementierung

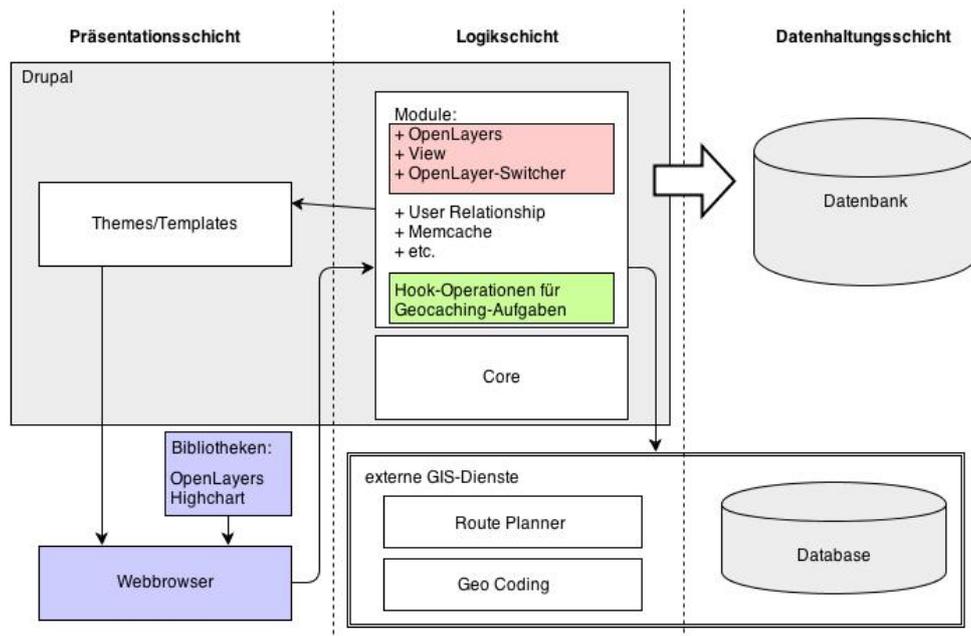


Abbildung 7.3: Das Websystem in Drei-Schichten-Architektur, bestehend aus der Präsentations-, der Logik- und der Datenhaltungsschicht.

7.3 Architektur und Aufbau des neuen Webportals

Der Aufbau des gesamten Web-Systems wurde als Drei-Schichten-Architektur, bestehend aus einer *Präsentations-*, einer *Logik-* und einer *Datenhaltungsschicht* umgesetzt. Mit Hilfe der Abbildung 7.3 wird auf die Komponenten im Einzelnen eingegangen:

- *Präsentationsschicht:* Diese Schicht ist verantwortlich, Informationen clientseitig darzustellen. Diese Daten werden von Drupal generiert und über ein Template in einem Browser angezeigt. Dieses Template ist verantwortlich für die Darstellung und Formattierung der Ausgabe. Zusätzlich zu Drupal wird die JavaScript *OpenLayers* und *Highcharts* Bibliothek eingebunden. Die Webseite, in welcher der Routenplaner eingebettet ist, wird zwar von Drupal aufgebaut und im Browser mit Hilfe eines Templates präsentiert, jedoch wird die Funktionalität des Routenplaners von extern liegenden *OpenLayers* übernommen. Die Speicherung der clientseitig erstellten und bearbeiteten Daten werden zur Gänze von Drupal übernommen.
- *Logikschicht:* Auf der mittleren Schicht werden die Aufgaben von Drupal abgehandelt. Basierend auf den Kernmodulen (Core) wird die Funktionalität von Drupal durch hinzugefügte Module erweitert. Wie bei einer WebGIS Architektur kann auf der mittleren Schicht auch die Geodatenverarbeitung gesehen werden. Diese externen Dienste übernehmen wichtige Berechnungen für das Portal, wie das Generieren von Streckenabschnitten für den Routenplaner oder Geocoding-Dienste. Angesprochen werden diese GIS Dienste mittels API Requests.
- *Datenhaltungsschicht:* Diese Schicht hält die Daten in einer MySQL Datenbank. Das Datenbankschema wird durch Drupal generiert und verwaltet.

Die folgende Beschreibung der Implementierung kann in drei Abschnitte unterteilt werden, welche in Abbildung 7.3 farblich hervorgehoben wurden:

1. *Verarbeitung und Präsentation der Streckendaten (blau)*: Dieser Abschnitt befasst sich damit, wie eine Strecke aus unterschiedlichen Quellen einer Drupal-Node hinzugefügt werden kann. Es wird beschrieben, wie diese Strecke mit Hilfe eines Routenplaners erstellt und mit *OpenLayers* angezeigt wird. Weiters wird gezeigt, wie eine bestehende Strecke aus der Datenbank oder aus einer GPX-Datei dargestellt werden kann.
2. *Umsetzung des Geocaching (grün)*: Die Implementierung von Geocaching zeigt die nötigen Berechnungen, um die Distanz zwischen geografischen Objekte herausfinden zu können. Durch *Hooks* muss im Drupal-System erkannt werden, wann eine neue Aktivität angelegt wurde. Die Strecke dieser Aktivität muss auf mögliche Geocaching-Objekte untersucht werden.
3. *Verwendung des OpenLayers-Moduls in Drupal 7 (rot)*: Die Verwendung des bereits vorhandenen *OpenLayer*-Moduls für Drupal in Verbindung mit dem *View*-Modul wird in diesem Abschnitt beschrieben. Dieses wird vor allem dazu verwendet, um alle Strecken des Portals auf einer Übersichtskarte präsentieren zu können. Zusätzlich wird gezeigt, wie mit dem Module *OpenLayers-Switcher* ein externer Layer-Switcher umgesetzt werden konnte.

Die Umsetzung und der Aufbau der komplexen Social-Community mit Hilfe von Drupal ist nicht Teil dieser Arbeit, sondern wird bei Sandriesser (2013) beschrieben.

7.4 Verarbeitung und Präsentation der Streckendaten

In dem Webportal wird es den Anforderungen entsprechend möglich sein, auf drei verschiedene Arten Streckendaten einzubringen: Über den Upload einer GPX-Datei, durch die Verwendung eines Routenplaners, sowie durch das Laden einer bereits bestehenden Strecke, welche zu einem früheren Zeitpunkt angelegt wurde. Im Drupal-System wird das Erstellen von Strecken innerhalb einer *Node* über ein *Template* eingebettet. So wird der Benutzer z. B. wie Abbildung 7.4 zeigt, eine neue sportliche Aktivität anlegen, was in Drupal dem Anlegen einer neuen *Node* entspricht.

Dieser Abschnitt befasst sich zunächst mit der Erstellung einer Route mit Hilfe eines Routenplaners. Anschließend wird auf die Auswertung einer GPX-Datei eingegangen. Zur Glättung dieser rohen Streckendaten aus einer GPX-Datei wird der Douglas-Peucker Algorithmus verwendet. Ebenso wie bei Strecken, welche aus der Datenbank geladen werden, können Streckeninformationen aus einer Datei nachträglich mit Hilfe eines Routenplaners geändert werden. Abschließend werden alle wichtigen Schritte näher erläutert, die die Berechnung der Streckendaten, wie z. B. Distanz oder Höhenveränderungen, übernehmen.

7.4.1 Erstellen einer Route durch Einbindung eines Routenplaners

Abbildung 7.5 zeigt den sequentiellen Ablauf des Erstellens einer Strecke. Bevor auf Details der Implementierung eingegangen wird, gibt dieses Diagramm einen groben Überblick. Zunächst wird das Platzieren eines Wegpunktes durch den User mit einem Klick auf die

7 Implementierung

The screenshot shows a web form for creating a new activity in Drupal. It is divided into several sections:

- Eine Aktivität ***: A text input field.
- Sportart ***: A dropdown menu with "Running" selected.
- Datum**: A date input field showing "26/06/2013", with a subtext "z.B. 26/06/2013".
- Zeit**: A time input field showing "10:30", with a subtext "z.B. 10:30".
- Route-Quellen**: Three buttons: "Erstelle neue Route" (highlighted), "Bestehende Strecke", and "Upload einer GPX-Datei".
- Start**: A button.
- Additional Options**: Three stacked buttons: "Füge Freunde hinzu", "Sichtbarkeit und Privatsphäre", and "Beschreibung der Aktivität".
- Speichern**: A final "Save" button.

Abbildung 7.4: Ansicht einer Node in Drupal, die dazu verwendet wird, eine neue Aktivität im System anlegen zu können. Die Route kann durch Auswahl einer der drei Quellen erstellt bzw. hinzugefügt werden.

Karte vorgenommen. Die Erstellung sowie Präsentation dieses Markers wird mit *OpenLayers* durchgeführt. Sobald zwei Marker vorhanden sind, können diese entweder direkt durch eine Linie verbunden werden, oder es wird serverseitig eine Route angefragt. Für alle Wegpunkte dieser Route müssen ebenso Höhendaten angefordert werden, bevor sie wieder zum Client zurückgesendet werden können. Dort werden diese Punkte wieder mit Hilfe von *OpenLayers* und einem *LineString*-Objekt auf der Karte präsentiert. Im nächsten Schritt werden alle wichtigen Daten zur Strecke berechnet, wie Distanz oder Höhenveränderung. Verändert der Benutzer die bereits vorhandenen Wegpunkte, z. B. durch Verschieben, so muss der Vorgang bei der Anfrage nach der Route wieder von neuem begonnen werden. Der gesamte Ablauf wiederholt sich, sooft neue Wegpunkte gesetzt werden.

Im Detail wird als erster wichtiger Teil das Platzieren von Wegpunkten auf einer Karte behandelt. Diese Wegpunkte können verschoben bzw. wieder gelöscht werden. Dies wurde mittels eines *Vector-Layers* implementiert. Serverseitig wird mit PHP die API Anfrage nach Routensegmenten und Höhendaten übernommen. Die Resultate dieser Anfrage werden anschließend wiederum durch *OpenLayers* auf der Karte präsentiert.

7 Implementierung

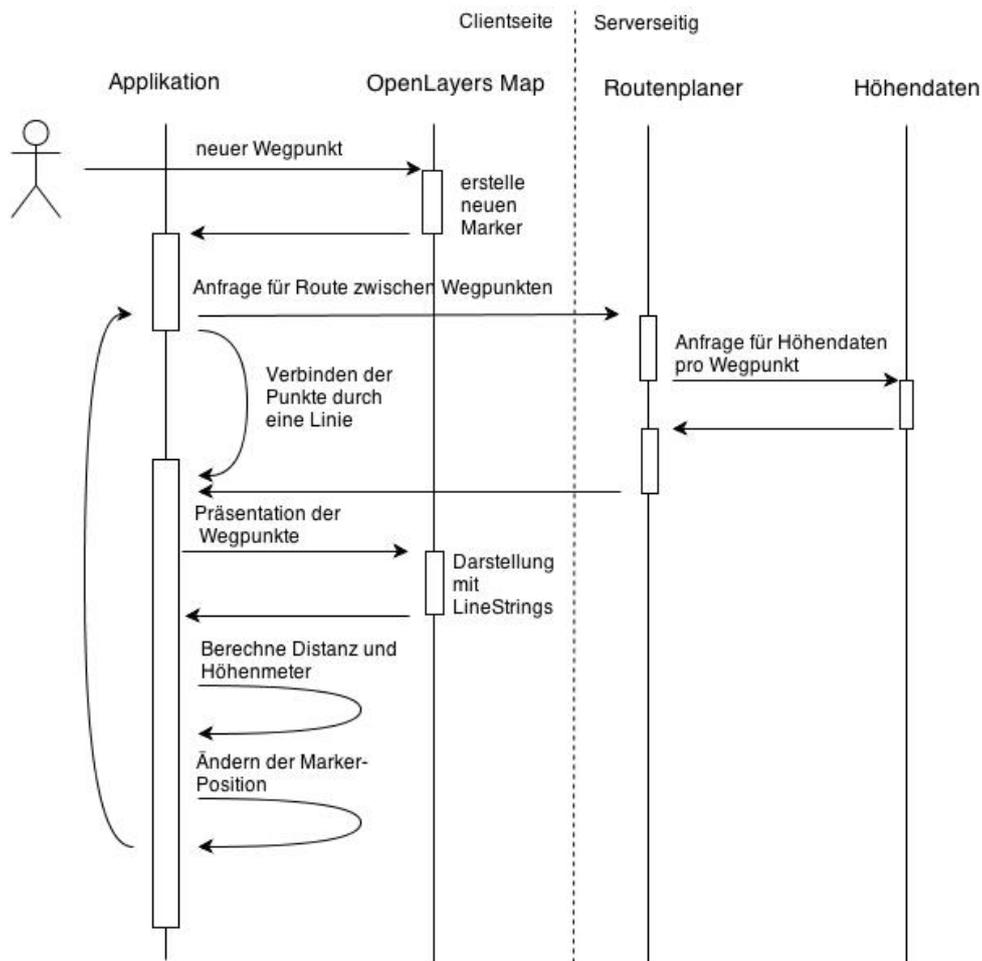


Abbildung 7.5: Das Sequenz-Diagramm zeigt den Ablauf beim Erstellen einer Strecke mithilfe eines Routenplaners. Clientseitig werden dem User Wegpunkte (Marker) sowie die Strecke auf einer OpenLayers-Karte angezeigt. Serverseitig müssen die Routen- sowie Höhendaten angefragt werden.

Platzieren und Verschieben von Wegpunkten

Der Planungsvorgang einer Strecke wird durch die Betätigung eines Button vom User aktiviert. Dadurch wird auf der OpenLayers Map ein Event registriert. Dieser Event fängt den Maus-Klick an einer beliebigen Stelle auf der Karte ab und erstellt ein Objekt, welches die aktuelle Position des Cursors, kommend vom Click Event, in Längen- und Breitengrade, hält. Exakt auf diesen Koordinaten wird ein Marker platziert. Dieser muss zumindest folgende Eigenschaften haben: Eine fortlaufende Nummerierung und die farbliche Abhebung jenes Markers, der gerade markiert ist. Dies wurde mit Hilfe eines Layer-Objekts umgesetzt, zu dem ein Feature mit einem Vector hinzugefügt wird.

Abbildung 7.6 zeigt einen Ausschnitt aus dem OpenLayers-Klassendiagramm (Schütze, 2008). Dargestellt werden jene Klassen, die in der Implementierung Anwendung gefunden haben. Zentral positioniert ist das Map-Objekt, zu welchem der Layer- sowie die Control-Objekte hinzugefügt werden. Durch das Control-Objekt können verschiedene Event-Listener, wie

7 Implementierung

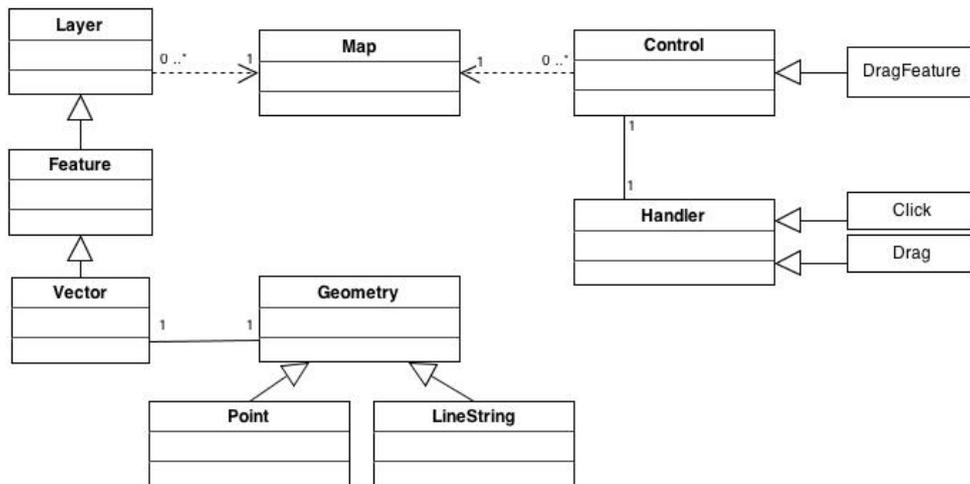


Abbildung 7.6: Dieser Ausschnitt aus dem *OpenLayers*-Klassendiagramm zeigt die, für die Implementierung wichtigen, Klassen. Das zentrale Map-Objekt hält zum einen die Layer-, zum anderen die Control-Objekte (Schütze, 2008).

der Click-Event, auf der Map, registriert werden. Der Layer wird durch ein Feature mit einem Vector definiert. Diesem Vector wird ein Geometry-Objekt hinzugefügt. Weil ein Marker erstellt wird, handelt es sich in diesem Fall um ein Point Objekt. Andere Geometry-Objekte wären z. B. MultiPoint Polygon oder LineString, welche an die *Simple Feature Access*-Spezifikation des OGC angelehnt sind. Dieses Point-Objekt dient als geometrische Basis für den Marker.

Zu diesem Zeitpunkt ist ein Objekt an der Stelle erstellt worden, wo ein Maus-Klick auf der Karte stattgefunden hat. Der Konstruktor des Vector-Objekts wird als Beispiel angeführt:

```

1 var viaPoint = new OpenLayers.Feature.Vector(point, null,
2   {externalGraphic: graphic,
3     graphicWidth: 18,
4     graphicHeight: 18,
5     fillOpacity: 1,
6     fillColor: "#000000",
7     fontOpacity: 1,
8     fontSize: "10px",
9     fontColor: fontcolor,
10    label: id.toString()
11  });

```

Listing 7.4: Erstellen eines Vector-Layers mit Angabe von Style-Informationen.

Listing 7.4 zeigt den Konstruktor eines Vector Objekts. Als erstes Attribut wird das Geometry-Objekt hinzugefügt (in diesem Fall als *point* definiert). Das zweite Attribut wird auf *null* gesetzt; hier bestünde die Möglichkeit weitere Attribute anzugeben. Innerhalb der geschwungenen Klammern wird der „Style“ definiert. Zur optisch besseren Darstellung wird eine Grafik eingebunden, die den Marker darstellt. Zusätzlich wird ein Label erzeugt, welches die Nummerierung der Marker übernimmt. Dieses Label kann mit den diversen Font Attributen gestaltet werden.

Im nächsten Schritt muss das Feature-Objekt auf der Karte präsentiert werden. Dies geschieht mit einem bereits beschriebenen Layer-Objekt. Zum besseren Verständnis wird die Verschachtelung von außen nach innen nochmal beschrieben: Die Map hält den Layer. In diesem Layer befindet sich das Feature-Objekt, welches das Geometry-Objekt hält. Damit ist die Präsentation eines Markers, der einen Wegpunkt darstellt, erfolgt. Es ist jedoch nicht nur möglich, neue Wegpunkte durch den Click-Event hinzuzufügen, sondern es können bereits erstellte Wegpunkte wieder verschoben oder gelöscht werden.

Das Verschieben der Wegpunkte wurde mit dem `Control.DragFeature` sowie dem `Handler.Drag` realisiert. Das `DragFeature` wird verwendet, um ein Element von `OpenLayers` mit der Maus bewegen zu können. In diesem Fall wäre es der `Layer`, in dem sich der Wegpunkt befindet. Dem Konstruktor des `Control`-Objekts wird der `Layer` übergeben. Weiteres werden darin die Eigenschaften definiert, die auf den `Layer` anwendbar sind. Diese Eigenschaften sind das `overFeature`, `outFeature` und `upFeature`.

- `overFeature`: Wird aufgerufen, sobald sich der Maus-Cursor über das `Layer`-Element, in diesem Fall der Wegpunkt, bewegt. Tritt dies auf, wird der Wegpunkt, in dem dessen `Feature` eine neue Grafik zugewiesen wird, markiert. Zusätzlich wird `HandlerDrag` aktiviert und der Maus-Cursor Style wird auf „Move“ gesetzt. Dadurch wird signalisiert, dass es möglich ist, den Wegpunkt zu verschieben.
- `outFeature`: Hier wird registriert, dass der Maus-Cursor den `Layer` wieder verlässt. Der Wegpunkt bleibt markiert, der Cursor muss jedoch vom „Move“ Style auf Normal zurückgesetzt werden.
- `upFeature`: Wird der Wegpunkt nach Verschieben wieder „losgelassen“, so wird dies von diesem `Feature` registriert. Nach dem Verschieben eines Wegpunktes muss die Route für den Abschnitt des Wegpunktes neu berechnet werden. Die neue Strecke würde dann von einem Wegpunkt davor bis zu einem Wegpunkt nach dem aktuell verschobenen beantragt werden.

Zusätzlich zu den Funktionen, die auf `OpenLayers` und der Karte zur Darstellung durchgeführt werden, müssen die Wegpunkt-Daten auch im System zwischengespeichert werden. Die einzelnen Wegpunkte werden in einem Array gespeichert, wobei ein Eintrag im Array einen Wegpunkt entspricht. Dieser Eintrag hält Informationen über Längen und Breitengrad, Höhenmeter, Zeit und eine ID. Im nächsten Schritt wird beschrieben, wie zunächst die Route zwischen zweier Wegpunkte abgefragt wird bzw. wie dann mit Hilfe eines `LineString` `Geometry`-Objektes die Strecke präsentiert wird.

Anfrage an die Routenplaner API und Darstellung der Route

Nach der Platzierung von zumindest zwei Markern, welche die Wegpunkte darstellen, muss der Weg zwischen diesen beiden Punkten bestimmt werden. Dies geschieht entweder durch Anfrage an eine Routenplaner API oder es wird direkt eine Linie zwischen den beiden Punkten gezeichnet. Die API benötigt Informationen von Längen- und Breitengrad der Start und Endpunkte. Diese werden mit Hilfe einer `OpenLayers` Koordinatentransformation berechnet. Nach dieser Berechnung werden sie zusammen mit den nötigen Attributen, wie Rückgabeformat oder Routenoptimierung für Fußgänger oder Radfahrer an die API übergeben.

7 Implementierung

Die nötigen Daten werden mittels *XHTMLRequest* an die serverseitige PHP Datei übergeben. Dort wird bei den eigentlichen Routenplaner Services angefragt. Um auf dem Portal immer einen Routenplaner anbieten zu können, werden zwei unterschiedliche Route-Services angesprochen, sollte einer der beiden ausfallen. Für die Verwendung des *OpenRouteService* war eine Anfrage bei den Entwicklern des ORS nötig. Durch Pascal Nies, einem der Programmierer, ergab sich die Möglichkeit, den Routenplaner über eine Service URL einbinden zu können. Gemäß der OpenLS Spezifikation des OGC kann diese URL via XML und HTTP POST angesprochen werden. Als Alternative wurde aufgrund seiner Zuverlässigkeit das Service von *Cloudmade* genommen, welches nach dem Anmelden über einen API-URL mit Identifikations-schlüssel angesprochen werden kann.

| ID | Längengrad | Breitengrad | Höhenmeter | Marker | Distanz |
|----|------------|-------------|------------|--------|---------|
| 0 | 15.411393 | 47.0507758 | 359 | 1 | 0 |
| 1 | 15.411390 | 47.050781 | 359 | 0 | 0.62 |
| 2 | 15.410860 | 47.050751 | 358 | 0 | 40.29 |
| 3 | 15.409740 | 47.050671 | 358 | 0 | 85.32 |
| 4 | 15.409730 | 47.050831 | 358 | 1 | 17.81 |
| 5 | 15.409740 | 47.050671 | 358 | 0 | 17.81 |
| 6 | 15.408770 | 47.050629 | 358 | 0 | 73.63 |
| 7 | 15.408220 | 47.050610 | 357 | 1 | 41.72 |
| 8 | 15.407520 | 47.050591 | 357 | 0 | 53.07 |
| 9 | 15.407630 | 47.049610 | 357 | 1 | 109.39 |

Tabelle 7.1: Dieses Array hält alle wichtigen Informationen pro Wegpunkt und gibt zusätzlich an, ob es sich bei dem Wegpunkt auch um einen Marker handelt.

Die Ergebnisse, die im XML Format zurückgegeben werden, müssen bereits serverseitig ausgewertet werden, da für jedes Koordinatenpaar auch Höheninformationen abgefragt werden müssen. Nachdem die Höhendaten pro Punkt vorliegen, werden sie an die Clientseite zurückgeschickt und präsentiert. Dort findet die Berechnung und Darstellung der Strecke statt. Die gesammelten Wegpunkte werden in ein Array gespeichert (siehe Tabelle 7.1). Dieses hält eine eindeutige ID, die Längen- und Breitengradinformationen, die Höhenmeter an der Stelle der Koordinate sowie die Distanz zwischen dem Punkt und dem vorherigen im Array.

Nachdem nicht jeder Wegpunkt der Strecke auch einem Marker entspricht, welcher verschoben oder gelöscht werden kann, ist nur in jenen Zeilen der Spalte *Marker* eine »1« gesetzt, bei denen ein solcher platziert wird. Veranschaulicht wird dies durch Abbildung 7.7, welche den Informationen von Tabelle 7.1 entspricht. Die vier ausgewiesenen Marker der Tabelle entsprechen den vier nummerierten Markern der Strecke. Alle anderen Punkte sind nicht erkennbare Wegpunkte, welche für die Definition der Strecke benötigt werden.

Die Präsentation der Strecke auf der Karte läuft ähnlich wie beim Erstellen eines Wegpunktes ab. Das Geometry-Objekt ist in diesem Fall ein *LineString*. Diese Geometrie präsentiert eine Linie und wird typischerweise bei der Markierung von Straßen verwendet (Jansen & Adams, 2010, S. 108). Basis des *LineString* bilden zwei *Point*-Objekte, welche die Start- und Endkoordinaten für die Linie darstellen.

7 Implementierung



Abbildung 7.7: Streckenabschnitt mit vier Wegpunkten. Zu erkennen ist, dass Wegpunkt 1 markiert ist.

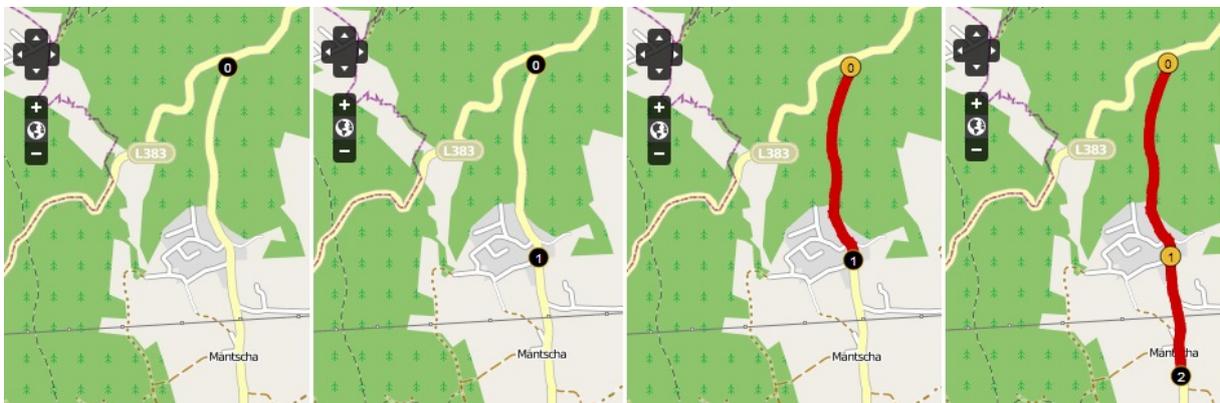


Abbildung 7.8: Der Ablauf der Routenerstellung zeigt nummerierte Wegpunkte, welche nacheinander gesetzt eine Route generieren.

Über eine Schleife werden aus allen Wegpunkten, die vom Routenplaner berechnet wurden und clientseitig im Array gespeichert sind, LineString-Objekte. Diese werden als Feature einem Layer hinzugefügt, welche auf der Karte die Strecke präsentieren. Werden weitere Routenpunkte hinzugefügt, so muss die Routenanfrage und Präsentation wiederholt werden. Die neuen Wegpunkte werden dem Array hinzugefügt und der Layer wird upgedatet. Über „Style“ Angaben lässt sich das Aussehen der Linien verändern, wie zum Beispiel Farbe oder Strichstärke. Abbildung 7.8 zeigt den Ablauf einer Routenerstellung. Nachdem zwei Wegpunkte gesetzt wurden, wird die Route dazwischen berechnet.

7.4.2 Präsentation von Daten einer GPX-Datei

Zusätzlich zum Erstellen einer Strecke im Routenplaner bietet das Webportal auch die Möglichkeit, GPS Informationen von einer anderen Quelle zu verwenden. Dies geschieht in Form einer GPX-Datei, die hochgeladen werden kann. In Folge des Parsens der Datei wird diese auf ihre

7 Implementierung

```
1 [...]
2 <trkseg>
3   <trkpt lon="15.4482781" lat="47.0710126">
4     <ele>368.86929321289</ele>
5     <time>2013-01-16T20:30:14.000Z</time>
6   </trkpt>
7   <trkpt lon="15.448240362114" lat="47.071011714365">
8     <ele>368.87057495117</ele>
9     <time>2013-01-16T20:30:28.000Z</time>
10  </trkpt>
11 </trkseg>
12 [...]
```

Listing 7.5: Dieser Ausschnitt aus einer GPX-Datei zeigt zwei Track-Point Elemente innerhalb eines Track-Segments.

Korrektheit überprüft. Listing 7.5 zeigt den für die Auswertung der Geoinformationen wichtigen Teil einer GPX-Datei. Es handelt sich um die gespeicherten Wegpunkte innerhalb eines `<trkseg>` Bereichs. Dieses Element enthält mindestens zwei Wegpunkte, die durch Längen- und Breitengrad und optionale Informationen wie Höhe und Zeit definiert werden.

Nach dem Parsen der Informationen hält das System ein Array mit Rohdaten, die möglicherweise zu viele Wegpunkte beinhaltet, von denen einige zur Berechnung von sportrelevanten Informationen überflüssig sind. Speziell die Ungenauigkeit des GPS Empfangs von bis zu zehn Metern kann dazu führen, dass die Wegpunkte in einem engen Bereich weit voneinander abweichen. Daher müssen diese Rohdaten der Wegpunkte geglättet werden. Zusätzlich führt diese Glättung auch zu einer Verringerung der zu verarbeitenden Datenmenge. Dieser Vorgang wird mit dem *Douglas Peucker Algorithmus* durchgeführt.

Der Douglas Peucker Algorithmus

Dieser Algorithmus wird für die Generalisierung und Kurvenglättung bei Vektorgrafiken verwendet, wenn eine Linie durch eine Punktfolge bestimmt wird. Seine Aufgabe ist es, Zwischenpunkte zu löschen und somit die Dichte der Koordinaten zu reduzieren, ohne aber dadurch die Charakteristik der Strecke zu verändern.

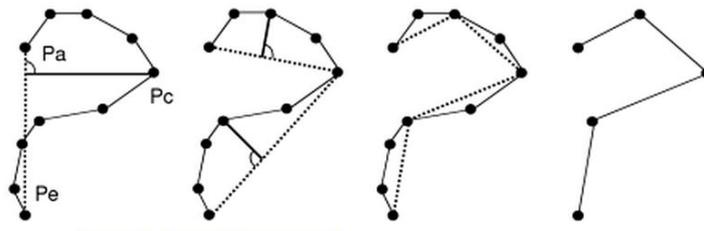


Abbildung 7.9: Schritte des Douglas Peucker Algorithmus zur Kurvenglättung (Lange, 2007, S. 90).

7 Implementierung

Abbildung 7.9 zeigt die einzelnen Schritte, die vom Algorithmus durchgeführt werden. Zuerst wird zwischen Anfangs- und Endpunkt eine Gerade P_aP_e gebildet. Anschließend wird eine Normale zum Punkt gelegt, die am weitesten von P_aP_e entfernt liegt, also P_c . Sollte dieser Abstand größer sein, als ein zuvor definierter Grenzwert, so bleibt der Punkt P_c erhalten. Rekursiv wird nun der Graph weiter untersucht, wobei die neuen Ausgangslinien zwischen P_aP_c und P_cP_e gebildet werden. Dies geschieht so lange, bis für alle Segmente der Abstand zum am weitesten entfernten Segmentpunkt unterhalb des Grenzwertes liegt. Je größer der Grenzwert, desto grober fällt die Glättung des Graphen aus (Lange, 2007, S. 91). Für *JavaScript* wurde eine Implementierung des *Douglas-Peucker-Algorithmus* von Adam Miller bereitgestellt, die unter der *Unlicense* steht (Adam, 2012).

| ID | Längengrad | Breitengrad | Höhenmeter | Zeit | Meter | Speed |
|----|-------------|-------------|------------|--------------------------|-------|-------|
| 1 | 15.41213036 | 47.04595566 | 356.75 | 2012-05-08T09:26:08.000Z | 0 | 0 |
| 2 | 15.41211891 | 47.04602051 | 359.48 | 2012-05-08T09:26:14.000Z | 7.26 | 4.36 |
| 3 | 15.41207027 | 47.04599762 | 356.52 | 2012-05-08T09:26:38.000Z | 4.48 | 0.67 |
| 4 | 15.4118166 | 47.04605865 | 353.47 | 2012-05-08T09:26:46.000Z | 20.38 | 9.17 |

Tabelle 7.2: Das Array, welches nach dem Parsen und Glätten der Wegpunkte aus der GPX-Datei alle wichtigen Informationen hält, unterscheidet sich vom Array der Routenplanung, weil zusätzliche Daten zu Zeit und Geschwindigkeit einfließen können.

Nachdem die Geoinformationen der GPX-Datei geglättet wurden, können sie auf der Karte präsentiert und berechnet werden. Tabelle 7.2 zeigt das Array, welches alle Wegpunkte, die die Strecke beschreiben, hält. Im Unterschied zum Array bei der Routenplanung (Tabelle 7.1) fließen hier Zeit und Geschwindigkeitsinformationen mit ein, sofern diese in der GPX-Datei gespeichert wurden. Die Präsentation der Strecke läuft identisch mit jener der Daten vom Routenplaner ab. Wiederrum wird in einem Layer-Objekt eine Feature eingebettet, die das geometrische *LineString*-Objekt, welches die Strecke präsentiert, hält.

Zusätzlich bietet das Portal nach der Präsentation aller relevanten Wegpunkte die Möglichkeit, diese Strecke direkt zu editieren und zu verändern. Dafür werden nach dem Betätigen des entsprechenden »Editier« Buttons Marker in regelmäßigen Abständen auf der Strecke platziert (siehe Abbildung 7.10). Diese Marker können verschoben, gelöscht oder neue Marker zwischen bereits existierenden hinzugefügt werden. Diese Funktionalität entspricht exakt jener bei der Erstellung einer Route mit Hilfe eines Routenplaners.

7.4.3 Präsentation einer bestehenden Strecke aus dem Drupal-System

Die dritte und letzte Möglichkeit, eine Route einer *Node* hinzufügen zu können, ist die Auswahl einer zu einem früheren Zeitpunkt erstellten Strecke. Mit einem Drop-Down-Feld werden dem User alle bereits erstellten Strecken präsentiert. Nachdem eine Strecke ausgewählt wurde, werden alle relevanten Daten zur Darstellung der Strecke aus der Datenbank geladen. Die weiteren Vorgänge entsprechen jenen der Präsentation von Informationen aus einer GPX-Datei: Die Wegpunkte, der bereits erstellten Strecke, werden präsentiert und können wiederum durch den User geändert werden. Wie in Abbildung 7.10 ersichtlich, werden zur Änderung des Streckenverlaufes Marker auf die Route platziert, welche verschoben oder gelöscht werden können.

7 Implementierung



Abbildung 7.10: Durch die Platzierung von Markern auf der Strecke kann das Streckenprofil einfach verändert werden.

7.4.4 Darstellung des Höhenprofils

Die Darstellung des Höhenprofils wurde clientseitig mit der *JavaScript* Bibliothek *Highcharts* umgesetzt. Damit ist es möglich, auf einfache Art effektvolle Diagramme zu erstellen. Übergeben werden muss das Array, welches die Daten der Wegpunkte hält. Durch Angabe der Werte, die auf der X- und Y-Achse präsentiert werden, wird das Diagramm gezeichnet. Im Höhendiagramm werden als X-Werte die Kilometer und als Y-Werte die Höhenangaben der einzelnen Wegpunkte verwendet. Zusätzliche Einstellungen ermöglichen das Erstellen eines Pop-ups, sobald der Maus-Cursor über das Diagramm fährt.

Eine Anforderung an die Darstellung der Strecke mit Höhendiagramm war, die Stelle auf der Strecke zu markieren, auf dem sich der Cursor auf dem Diagramm befindet, et vice versa. Da diese beiden Möglichkeiten sehr unterschiedlich umgesetzt werden mussten, werden sie in der Folge einzeln beschrieben. Abbildung 7.11 zeigt das Resultat der Verbindung von der Präsentation der Strecke in der Karte sowie dem Höhenprofil als Diagramm.

Cursor auf dem Diagramm-Highlight des Streckensegments

Als *tooltip* wird bei *Highcharts* jenes Pop-up bezeichnet, welches erscheint, sobald der Cursor über dem Diagramm gesetzt wird. Im Konstruktor des Diagramms kann der gewünschte Inhalt dieses *tooltip* angegeben werden. Es werden die Streckenkilometer sowie die Höhe an diesem Punkt in Pop-ups ausgegeben. Zusätzlich wird eine Funktion aufgerufen, die das Streckensegment an der richtigen Stelle auf der Karte markiert. Dieser Funktion wird die ID des aktuell auf dem Diagramm markierten Wegpunktes mitgegeben. In der Funktion wird ein neuer Layer mit einem *LineString* erstellt. Dieser Layer muss sich farblich von dem Strecken-Layer abheben, um ein Highlight simulieren zu können. Sobald der Cursor das Diagramm verlässt, verschwindet das *tooltip* sowie die Markierung auf der Strecke. Dies geschieht durch das Löschen des neuen Layer-Objekts beim *MouseOut*-Event in der *Highchart*.

7 Implementierung

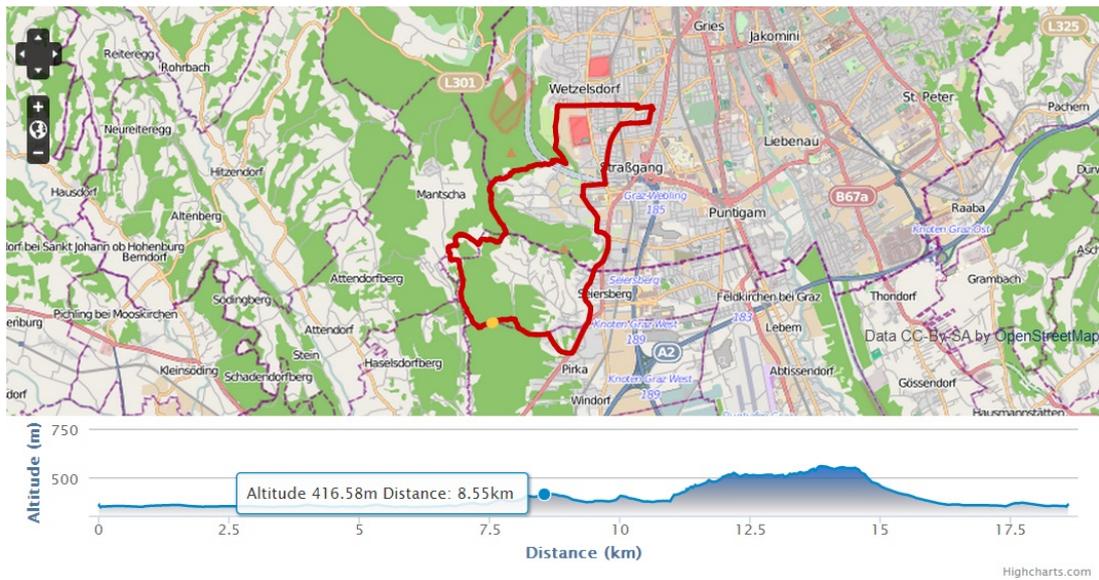


Abbildung 7.11: Der gelbe Abschnitt auf der Strecke entspricht dem Punkt im Diagramm, bei welchem das Pop-up angezeigt wird.

Cursor über der Strecke-Highlight des Diagramms mit Pop-up

Wird der Cursor über die Strecke auf der Karte bewegt, so führt auch dies zu einem Highlight des Streckensegments sowie zu einem Pop-ups auf dem Höhendiagramm. Um das umzusetzen, muss der Layer, welcher die Strecke hält, mit zusätzlicher Funktionalität ausgestattet werden. Es ist nötig, das Control-Element `SelectFeature` auf dem Layer zu definieren. Dies erlaubt, den `default-Style` des Streckensegments in einen `temporary-Style` zu ändern, der eine farbliche Änderung ermöglicht. Zusätzlich wird das `Highchart` Diagramm angesprochen und das `Tooltip` Element an der korrekten Stelle im Diagramm aufgerufen. Beim Verlassen des Cursors der Strecke wird der Style wieder auf `default` geändert und das Pop-up auf dem Diagramm entfernt.

7.4.5 Berechnung von Distanz, Höhe und Geschwindigkeit

Nachdem die Darstellung der Routen für Quellen aus einer GPX-Datei, des Routenplaners oder einer bestehenden Strecke gezeigt wurde, wird nachfolgend auf die Berechnung von Distanz, Geschwindigkeit und Höhenveränderung eingegangen. Diese Berechnungen werden unabhängig von der Quelle der Route durchgeführt. Beim Berechnen der Distanz muss jedes Segment zwischen zwei Wegpunkten berechnet und anschließend aufsummiert werden. Die Berechnung zwischen den Punkten wird mit sphärischer Trigonometrie durchgeführt, womit eine exakte Entfernungsberechnung auf einer Kugel möglich ist. Abbildung 7.12 zeigt die schematische Darstellung der relevanten Winkel und Koordinaten (Kompf, 2013).

```
1 distance = 6378.388*acos(sin(lat1)* sin(lat2)+cos(lat1)*cos(lat2)*cos(lon2 -lon1));
```

Listing 7.6: Berechnung der Distanz zwischen Koordinaten

7 Implementierung

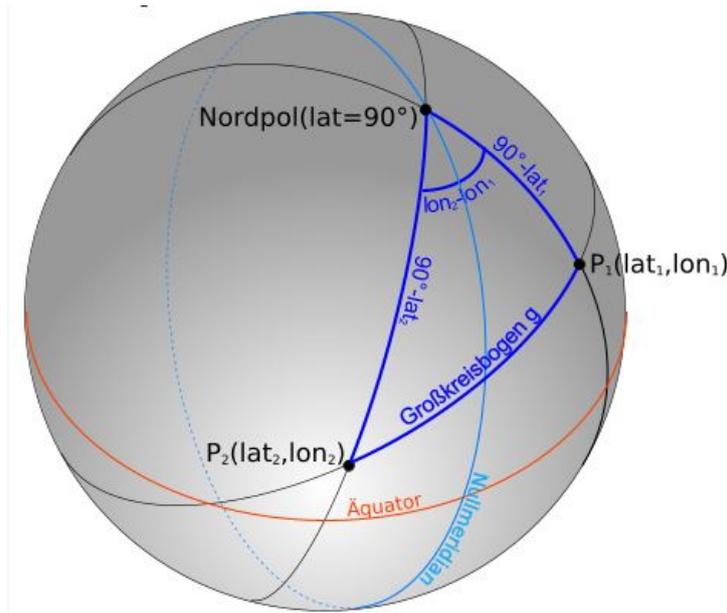


Abbildung 7.12: Darstellung der Distanzberechnung mit sphärischen Koordinaten. Diese Berechnung auf der Kugeloberfläche erfolgt mit Hilfe des Kugeldreiecks (Kompf, 2013).

Der Nordpol, P_1 und P_2 bilden ein Dreieck auf der Kugel. Bekannte Größen sind die zwei Seiten P_1 Nordpol und P_2 Nordpol und deren eingeschlossener Winkel $lon_2 - lon_1$. Mit Hilfe des Seitenkosinussatzes und den Multiplikation des Erdradius wird die Formel aus Listing 7.6 hergeleitet. Sie zeigt die Berechnung der Distanz zweier Koordinaten, wobei lat und lon für die jeweiligen Längen- und Breitengrade im Radianten steht. Multipliziert mit dem vereinfachten Erdradius wird der Abstand in Kilometer berechnet. Durch die Berechnung wird die jeweilige Distanz zwischen den Punkten sowie durch das Aufsummieren die Gesamtdistanz berechnet.

Die Segment-Distanz wird benötigt, um die Geschwindigkeit zwischen den jeweiligen Punkten zu bestimmen. Dies kann jedoch nur bei den Daten aus einer GPX-Datei erfolgen, die einen Timestamp bei jedem Wegpunkt beinhaltet. Durch Berechnung der Delta-Zeit zwischen zwei Wegpunkten und dem Wissen der Entfernung dieser kann die Durchschnittsgeschwindigkeit für das Segment bestimmt werden. Zusätzlich lassen sich Höchstgeschwindigkeit sowie der Durchschnitt über die gesamte Strecke bestimmen. Bei Strecken, die mit dem Routenplaner erstellt wurden, kann durch Angabe der Gesamtdauer nur eine Durchschnittszeit berechnet werden. Die Höheninformationen werden ebenfalls durch Aufsummieren bzw. Abziehen der Änderungen der Höhenwerte berechnet. Relevante Informationen sind hier die Gesamtsummen der positiven Höhenänderungen, der negativen Höhenänderungen, sowie der höchste Punkt der Strecke.

7.4.6 Speicherung der Daten im Drupal-System

Nachdem alle Informationen die Strecke betreffend clientseitig vorhanden sind, müssen diese in Felder der Drupal-Node geschrieben werden. Dazu wird der Inhalt des Arrays, der die

Wegpunkte hält, als String in das entsprechende Feld geschrieben. In Integer-Felder werden die berechneten Daten wie z. B. Distanz oder Höhenmeter eingetragen. Diese Felder werden durch das Template versteckt und sind somit für die Benutzer nicht sichtbar. Beim Abspeichern der *Node* werden alle Felder und deren Daten von Drupal automatisch in die Datenbank geschrieben. Dort wird für jedes Feld einer *Node* eine eigene Tabelle angelegt.

7.5 Umsetzung des Geocaching

Der erste wichtige Teil der Funktionalität wurde mit der Verarbeitung und Präsentation von Streckendaten aus internen oder externen Quellen beschrieben. Die weiteren Anforderungen an das Portal waren eine innovative und neue Mischung von sportlichen Geodaten mit Geocaching-Elementen. Die Implementierung der unterschiedlichen Geocaching-Funktionalität wird im folgenden Abschnitt beschrieben. Unterteilt wird hier in Punkt-Objekte, Strecken-Objekte sowie virtuellen Rennstrecken, die mittels der Strecke einer sportlichen Aktivität aufzusammeln sind.

7.5.1 Erstellen eines Geocaching-Objekts

Das Erstellen von Geocaching-Objekten ist prinzipiell mit dem Prozess der Erstellung einer Strecke für eine Aktivität identisch. Ein Strecken-Objekt oder eine virtuelle Rennstrecke wird mit Hilfe des Routenplaners erstellt. Hier wird dieselbe Funktionalität angeboten, als würde eine Strecke für eine Aktivität erstellt werden. Ist das gewünschte Strecken-Objekt vorhanden, so wird zusätzlich ein Grenzbereich um die Strecke definiert. Die Seiten dieses Bereiches liegen auf den Punkten der Strecke, welche am weitesten nördlich, südlich, westlich und östlich liegen. Dies wird später dabei helfen, effizient nach Geocaching-Objekten zu suchen. Alternativ zu einer Strecke kann ein Punkt-Objekt erstellt werden. Hier wird lediglich ein Punkt auf der Karte definiert. Dieser kann, wie bei den Wegpunkten einer Strecke, verschoben oder gelöscht und neu gesetzt werden.

7.5.2 Ablauf des Geocachings

Der Ablauf im System wird in Abbildung 7.13 dargestellt und kann wie folgt beschrieben werden:

1. Eine neue Aktivität mit Strecke wird angelegt.
2. Zusätzlich zur Strecke werden die nord-, süd-, west- und östlichen Streckenextreme gespeichert.
3. Nach dem Speichern der Aktivität muss im System nach Caching-Objekten innerhalb der Streckenextreme gesucht werden.
4. Alle Objekte im definierten Bereich müssen einzeln überprüft werden, ob sie innerhalb einer Toleranzentfernung zur Strecke liegen.
5. Wird die Toleranz nicht überschritten, so gilt das Geocaching-Objekt als aufgesammelt.
6. Handelt es sich um eine virtuelle Rennstrecke, muss zusätzlich die benötigte Zeit für das Segment berechnet werden.

7 Implementierung

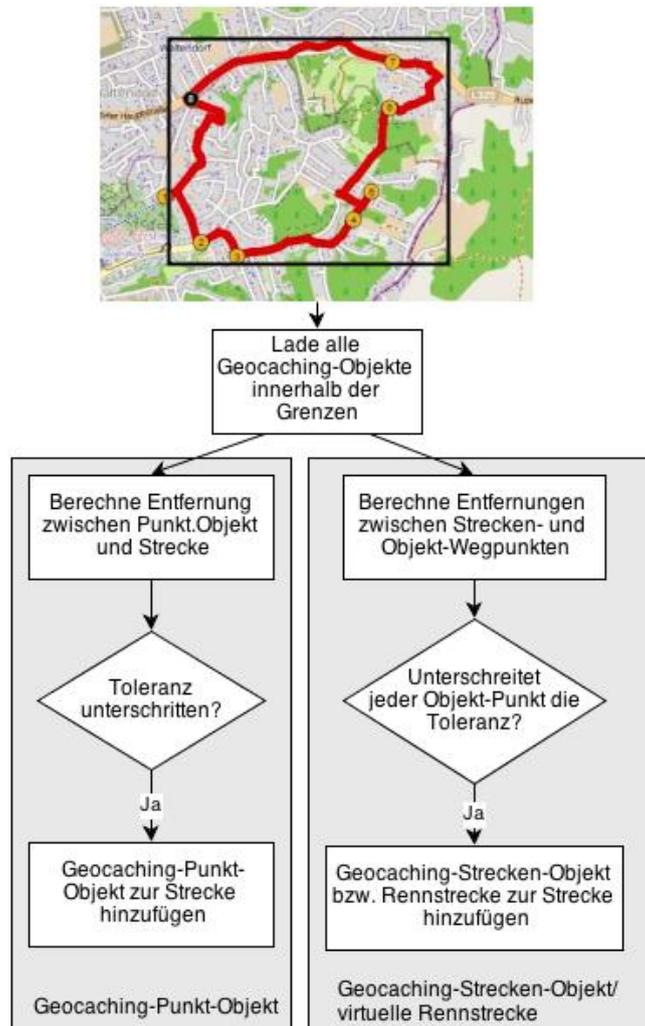


Abbildung 7.13: Wird innerhalb der Grenzen der Strecke ein Geocaching-Objekt gefunden, so muss bei der Berechnung zwischen einem Punkt- und einem Strecken-Objekt unterschieden werden.

Gleich wie bei dem Geocaching-Objekt werden auch bei der Strecke der Aktivität die Extreme gespeichert. In Drupal wird mittels eines *Hooks* das Abspeichern einer neuen oder geänderten Aktivität abgefangen. Wird ein solcher Vorgang registriert, müssen jene *Nodes*, also Geocaching-Objekte aus der Datenbank mittels *mysql* geladen werden, die kleiner oder gleiche Extreme in alle vier Haupthimmelsrichtungen haben, wie die Strecke der Aktivität. Dadurch können alle in Frage kommenden Objekte im Bereich der Strecke geladen werden. Anschließend muss für alle Objekte die Entfernung zur Strecke berechnet werden und handelt es sich um eine Rennstrecke muss zusätzlich noch die Zeit berechnet werden.

7.5.3 Berechnung der Entfernung von geografischen Objekten zueinander

Im System müssen zwei verschiedene Arten von Geocaching-Objekten mit Strecken kontrolliert werden: Zum einen die Punkt-Objekte, zum anderen die Strecken-Objekte. Nach-

dem die virtuellen Rennstrecken ebenso Strecken-Objekte sind und sich diese nur in der zusätzlichen Berechnung der Zeit unterscheiden, läuft der Kontrollalgorithmus gleich wie jener des Strecken-Objekts ab. Die Implementierung wurde mit Trigonometrie und dem Satz des Heron nach einer Vorlage von Kamm² umgesetzt.

Geocaching eines Punkt-Objekts

Ein Punkt-Objekt wird dann von einer Strecke eingesammelt, wenn die Koordinaten des Objektes an irgendeiner Stelle der Strecke innerhalb einer Toleranzgrenze liegen. Am einfachsten lässt sich diese Berechnung durch Einsatz von Trigonometrie durchführen. Die Strecke besteht im Prinzip aus einer Abfolge von Wegpunkten, die verbundene Streckensegmente ergeben. Es lässt sich pro Streckensegment ein Dreieck mit dem Geocaching-Punkt-Objekt bilden. Folglich wird mit jedem Streckensegment, welches in Summe die Strecke beschreibt, ein Dreieck mit dem Punkt-Objekt gebildet um daraus die Distanzen der Streckensegmente zum Punkt zu berechnen. Ist eine dieser berechneten Distanzen kleiner als eine definierte Toleranz, so liegt der Punkt auf der Strecke.

Dieses Dreieck wird durch die Punkte A und B , die zwei Wegpunkte der Strecke und somit ein Streckensegment darstellen und dem Punkt C , welcher das Punkt-Objekt darstellt, beschrieben. Zunächst werden die einzelnen Seiten des Dreiecks (AB, AC und BC) und deren innerer Winkel berechnet. Weist ein Winkel 0° oder mehr als 90° auf, so liegt der Punkt entweder direkt auf der Linie, oder eine der beiden Seiten AC oder BC bestimmt bereits die Entfernung zwischen den beiden Objekten.

Mit dem Satz des Heron lässt sich mit den Seiten und dem halben Umfang s die Fläche des Dreiecks mit Hilfe folgender Formeln berechnen:

$$s = \frac{1}{2} (a + b + c)$$

$$A = \sqrt{s(s-a)(s-b)(s-c)}$$

Aus der Fläche lässt sich die Höhe des Dreiecks bzw. der Abstand des Punktes C von der Linie AB berechnen:

$$h = \frac{2A}{a}$$

Ist dieser Abstand größer als die Toleranzgrenze so muss mit dem nächsten Streckensegment dieselbe Berechnung durchgeführt werden. Erfüllt keine Distanz die Toleranz, so liegt das Punkt-Objekt nicht auf der Strecke.

Abbildung 7.17 versucht schematisch diese Berechnung darzustellen. Definiert durch die Wegpunkte A, B, C, D, E und F werden die Liniensegmente L_1, L_2, L_3, L_4 und L_5 aufgespannt.

²<http://www.sitepoint.com/forums/showthread.php?661462-Distance-between-long-lat-point-and-line-segment> – zuletzt abgerufen am 10.05.2013.

7 Implementierung

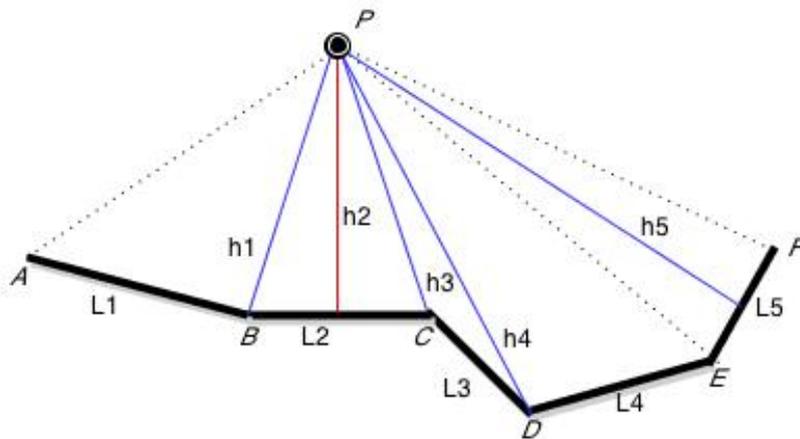


Abbildung 7.14: Berechnung des Abstands eines Punkt-Objekts zu einer Strecke.

Der Punkt P repräsentiert das Geocaching-Objekt. Im ersten Schritt wird das Dreieck ABP untersucht. Nachdem einer der Winkel größer ist als 90° , ist die Distanz zum Punkt P die Strecke BP . Das zweite Dreieck, BCP , muss über den Satz des Heron berechnet werden, um auf die Höhe $h2$ zu kommen. Dieser Vorgang wird so oft wiederholt, bis alle Streckensegmente untersucht wurden. Die kürzeste Entfernung zwischen Strecke und Punkt-Objekt wäre in diesem Fall der Abstand $h2$.

Geocaching eines Strecken-Objekts

Die Kontrolle, ob ein Geocaching-Strecken-Objekt mit einer Strecke übereinstimmt, läuft ähnlich jenes eines Punkt-Objekts ab. Nachdem alle zutreffenden Strecken-Objekte, die im bestimmten Bereich vorhanden sind, mit MySQL aus der Datenbank geladen wurden, wird zuerst kontrolliert, ob die Distanz des Geocaching-Objektes kleiner ist als jene der Strecke, da eine längere Distanz der Objekte zu keiner Übereinstimmung führen würde. Ist das Strecken-Objekt maximal ein Teil der Strecke der Aktivität, so kann mit der Kontrolle begonnen werden.

Der Kontrollalgorithmus entspricht jenem des Punkt-Objekts. Jene Wegpunkte, die das Strecken-Objekt definieren, werden einzeln mit allen Segmenten verglichen, welche die Strecke der Aktivität aufspannen. Sobald eine Distanz zwischen Wegpunkt des Geocaching-Objekts und Streckensegment nicht der Toleranzgrenze entspricht, kann die Kontrolle abgebrochen werden, da für einen positiven Ausgang der Kontrolle das gesamte Strecken-Objekt die Toleranz erfüllen muss.

7.5.4 Berechnung von Zeitintervallen auf Streckensegmenten

Stimmt die Strecke einer Aktivität mit einer virtuellen Rennstrecke überein, so muss zusätzlich die Zeitdauer, die für das Segment benötigt wurde, berechnet werden. Dafür wird zunächst der erste Wegpunkt der Rennstrecke mit der Strecke verglichen. Von dem Segment, welches

7 Implementierung



Abbildung 7.16: Auf der Übersichtskarte wird der Streckenverlauf in Verbindung mit einem Pop-up angezeigt.

Darstellung verschiedener Views in einer Map

Werden Geodaten mit *OpenLayers* präsentiert, müssen diese mit dem *Views* Modul bereitgestellt werden. *Views* gilt als eines der am meisten verwendete Module in Drupal (Miles, 2013). Es erlaubt einfaches und schnelles generieren von Informationen, welche z. B. Seiteninhalte oder Benutzer sein können. Ein *View* stellt eine einfache Form einer SQL-Datenbank Abfrage dar. Durch die angegebenen Filter wird bestimmt, welcher Inhalt bereitgestellt wird. Abbildung 7.17 zeigt einen *OpenLayers* Block, mit Markern, welche die Ausgabe der *Views* darstellen, und den externen Layer Switcher.

Für jede Art von Inhalt, die auf der *OpenLayers* Karte präsentiert wird, muss eine eigene View erstellt werden. Es wird zunächst unterschieden in sportliche Aktivitäten, wie z. B. Laufen, Rennrad fahren oder Mountainbiken, sowie in Geocaching-Objekte, wie Punkt- oder Strecken-Objekte bzw. virtuelle Rennstrecken. Bei den *OpenLayers* Einstellungen lassen sich diese verschiedenen *Views* in die Karte integrieren. So werden z. B. alle Laufaktivitäten, die im System vorhanden sind, auf der Karte präsentiert. Dies geschieht durch Platzierung eines Markers an deren Startkoordinaten. Zusätzlich erlauben die richtigen Einstellungen das Erscheinen eines Pop-ups bei der Auswahl eines Markers. Mit Hilfe dieses Pop-ups wird die ausgewählte Aktivität durch deren Namen, Distanz sowie deren Ersteller näher beschrieben.

Mit Erscheinen des Pop-ups wird auch der Streckenverlauf auf der Karte angezeigt. Dies war lediglich durch Einstellungen in der Backend-Umgebung von Drupal nicht realisierbar und musste daher implementiert werden. Dafür wurde im *OpenLayers*-Modul Ordner die `openlayers_behavior_Pop-ups` *JavaScript* Datei bearbeitet. Diese regelt das Erscheinen, Aussehen und Verhalten des Pop-ups. Im Zuge dieses Pop-ups wurden die Geoinformationen der Strecke in die Karte geladen. Ähnlich wie bei der Präsentation der Informationen aus einer GPX-Datei wurde auch hier ein Layer angelegt, welcher mit `LineStrings` die einzelnen Wegpunkte der Strecke verbindet (siehe Abbildung 7.17). Abschließend wurde mit der Funktion `zoomToExtent` der angezeigte Kartenbereich auf die Strecke optimiert. Bei der Deaktivierung des Pop-ups wurde dieser Layer wieder gelöscht. Da es vorkommen kann, dass das Pop-up über der angezeigten Strecke erscheint, wurde die Möglichkeit implementiert, das Pop-up per

7 Implementierung

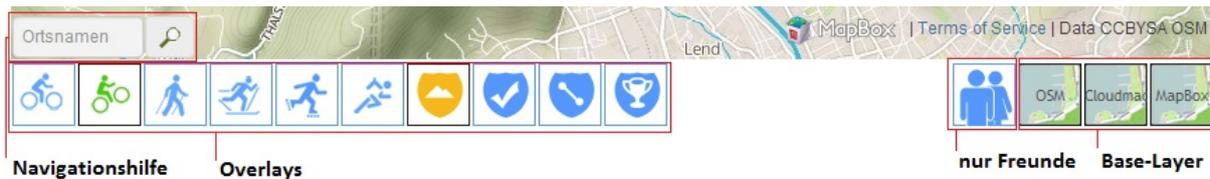


Abbildung 7.17: Mit dem externen LayerSwitcher ist es möglich, einfach die dargestellten Sportarten ein- und auszublenden. Zusätzlich gibt es eine Navigationshilfe, die ein schnelles Navigieren zu einem Ort ermöglicht.

Maus zu verschieben. Dies ermöglicht das Pop-up dort zu platzieren, wo es keine wichtigen Informationen verdeckt.

Umsetzung eines extern liegenden LayerSwitcher

Bereits am Anfang der Implementierungsphase wurde klar, dass der Standard-LayerSwitcher von *OpenLayers* grafisch nicht sehr ansprechend wirkt, weshalb nach einer Alternative dazu gesucht wurde. Im Zuge dieser Recherche wurde ein Modul³ für Drupal gefunden, welches es erlaubt, den LayerSwitcher extern in einen *Block* zu positionieren. Dieser *Block* lässt sich dann je nach Designvorstellung platzieren. Dieser externe LayerSwitcher ermöglicht es, pro Base bzw. Overlay Layer eigene Styles zu definieren, um grafisch ansprechende Effekte zu erstellen.

Dieses Modul wurde im Zuge der Implementierung angepasst und erweitert. Dadurch wurde es möglich, die Layer so zu ändern, dass auf Wunsch nur noch Inhalte von „Freunden“ aus der Social-Web-Community auf der Karte angezeigt werden. Wie bereits erwähnt, wird pro angezeigten Inhalt ein *View* verwendet. Es wird nun jeweils ein eigenes *View* benötigt, für z. B. „alle Lauf-Aktivitäten“ bzw. „nur Lauf-Aktivitäten meiner Freunde“. Bei letzterem wurde im *View* zusätzlich ein Filter aktiviert, der nur Inhalte von Freunden durchlässt. Möchte der Benutzer nur noch Inhalte seiner Freunde sehen, so werden nur noch die Layer geladen, die im *View* den Freunde-Filter integriert haben, und die anderen Layer werden entfernt.

Abbildung 7.17 zeigt den extern liegenden LayerSwitcher. Dieser besteht aus zwei Teilen. Links sind die verschiedenen Overlays zu erkennen, bestehend aus den Sportarten und den Geocaching-Objekten und rechts befinden sich zwei BaseLayer (OpenStreetMaps und Bing Satelite). Weiters ist eine zusätzliche Navigationshilfe ersichtlich. Diese arbeitet mit Geocoding um die Karte an eine eingegebene Adresse zu zoomen.

7.7 Zusammenfassung

Zusammenfassend zeigte dieses Kapitel, welche Schritte unternommen wurden, um die gefundenen Anforderungen umsetzen zu können. Die Architektur dieser Web-Applikation ähnelt jener einer klassischen WebGIS Anwendung, die auf Basis von Drupal umgesetzt wurde.

³https://github.com/LambethCouncil/opencouncil/blob/master/sites/all/modules/custom/openlayers_switcher/openlayers_switcher.module – zuletzt abgerufen am 13.05.2013.

7 Implementierung

Durch den Einsatz von Drupal wurde es möglich, ein umfangreiches Community-Websystem in effizienter Zeit aufzubauen. Dies geschah hauptsächlich durch den Einsatz externer Module, welche den Funktionsumfang erweiterten.

Der Einsatz von *OpenLayers* fand einerseits extern, lediglich eingebettet in eine Drupal-Template, andererseits intern, integriert als Modul, statt. Da die Funktionalität des Routenplaners abgekoppelt von jener von Drupal arbeitet, wurde dieser lediglich in eine Template eingebettet. Durch clientseitige Eingabe von Daten am Routenplaner (Setzen von Wegpunkten), wurde serverseitig nach GIS Daten via API angefragt. Das *OpenLayers* Drupal Modul wurde für die Übersichtskarte verwendet, da mit Hilfe von *Views* die Präsentation von Geodaten aus dem System einfach möglich ist.

Um die Geocaching-Funktionalität umsetzen zu können, musste über einen *Hook* eine neu erstellte Aktivität abgefangen und auf Geocaching-Objekte überprüft werden. Die einzelnen Schritte, wie der Abstand eines Punkt- oder Strecken-Objektes zur Strecke berechnet werden kann, wurde mit Hilfe von Trigonometrie gezeigt. Dadurch wurden alle Punkte, die in den Anforderungen vorgegeben wurden, umgesetzt. Die Verwendung der daraus resultierenden Funktionalität wird im nächsten Kapitel gezeigt.

8 Analyse aus Benutzersicht

Diese Analyse zeigt den Ablauf der in der Implementierung erstellten Funktionalität. Begonnen wird dieses Kapitel zunächst mit der Übersichtskarte aller Geodaten. Diese wird auf der Startseite des Webportals präsentiert. Auf dieser Karte sind alle Aktivitäten, Geocaching sowie Rennstrecken-Objekte verzeichnet. Des Weiteren wird in diesem Abschnitt gezeigt, wie eine neue Aktivität angelegt werden kann. Dafür wird eine Strecke mit dem Routenplaner erstellt und zusätzlich die Änderungsmöglichkeiten für bestehende Strecken aus einer GPX-Datei oder aus dem System beschrieben. Da Geocaching-Objekte nach demselben Ablauf erstellt werden, wird die Aktivität auch repräsentativ für diese Objekte gesehen. Der zweite Teil dieses Abschnitts geht auf die Präsentation sowie das Einsammeln dieser Geocaching-Objekte ein.

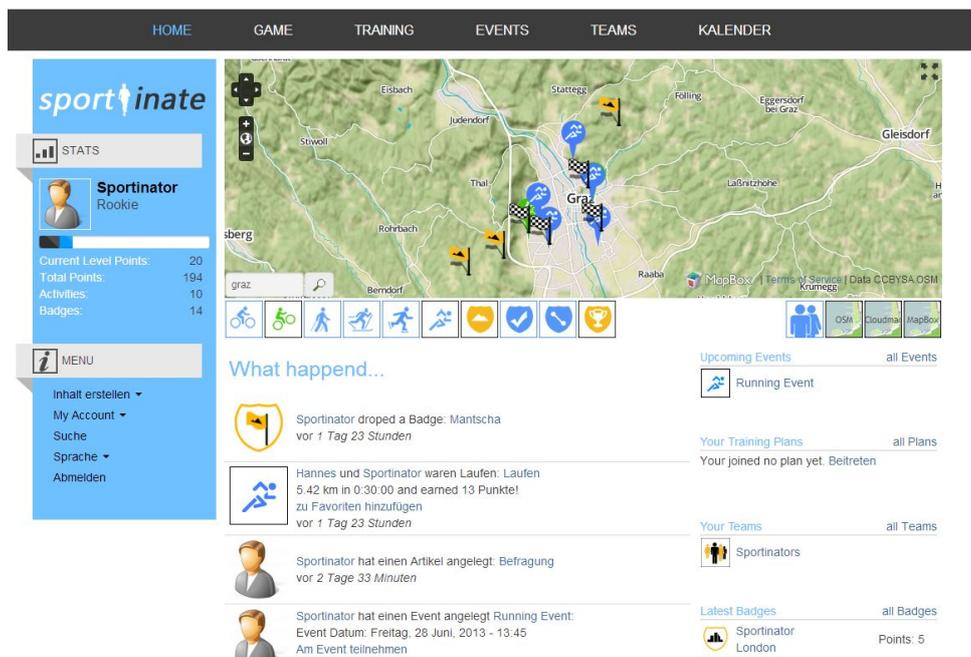


Abbildung 8.1: Ansicht der Startseite nach dem Einloggen. Zentrales Objekt ist die Übersichtskarte mit dem extern liegenden Layer-Switcher.

8.1 Die Übersichtskarte

Abbildung 8.1 zeigt die Startseite des Webportals. Hauptaugenmerk wird hier auf die Darstellung der Übersichtskarte gelegt. Dem Benutzer zeigt sie die Position der verschiedenen

8 Analyse aus Benutzersicht

Geodaten-Objekte. Welche angezeigt werden, kann mit Hilfe des Layer-Switchers eingestellt werden. Dieser unterteilt sich in die Sportarten, die Geocaching-Objekte sowie die Rennstrecken. Zusätzlich hat der User die Möglichkeit, zwischen drei verschiedenen Basiskarten zu wählen. Die verschiedenen Sportarten werden mit einem abgerundeten Pin und dem jeweiligen Logo der Sportart dargestellt. Bei einem Klick auf diesen Pin, wird ein Pop-Up geöffnet, in dem der Name der Aktivität sowie der Ersteller beschrieben wird. Optional kann auch die Streckenführung angezeigt werden. Befinden sich mehrere Aktivitäten gleicher Sportart sehr nah bei einander, so werden diese in einem Pin zusammengefasst. Wird dieser ausgewählt, werden alle Aktivitäten in einer Liste untereinander angezeigt, was sie einzeln auswählbar macht.

Geocaching-Objekte werden in Form von gelben Fahnen mit unterschiedlichem Logo angezeigt. Die Punkt-Objekte unterscheiden sich in Berggipfel- und Standard-Objekte. Handelt es sich um ein normales Streckenobjekt, so wird dieses auch durch eine gelbe Fahne mit einer Strecke als Logo angezeigt. Rennstrecken werden mit einer schwarz-weiß karierten Fahne dargestellt.

The screenshot shows the 'sportinate' web portal interface for creating a new activity. The top navigation bar includes 'HOME', 'GAME', 'TRAINING', 'EVENTS', 'TEAMS', and 'KALENDER'. The left sidebar contains the user profile for 'Sportinator' (Contestant) with statistics: 'Aktuelle Punkte: 32', 'Alle Punkte: 418', 'Aktivitäten: 17', and 'Badges: 28'. The main form area is titled 'Eine Aktivität *' and includes the following fields and controls:

- Eine Aktivität ***: Text input field.
- Datum**: Date input field with the value '25/07/2013' and a note 'z.B. 25/07/2013'.
- Sportart ***: Dropdown menu with 'Running' selected.
- Zeit**: Time input field with the value '12:00' and a note 'z.B. 12:00'.
- Buttons**: 'Start', 'Beenden', 'Marker hinzufügen', and 'Lösche Marker'.
- Wähle einen Routenplaner:** Radio buttons for 'CloudMade' (selected), 'ORS (Nur in Europa verfügbar)', and 'Einfache Linie'.
- Dauer:** Input field with '00:30:00 (h/m/s)'.
- Map:** A map showing a route in a city area with labels like 'Wetzelsdorf', 'Belgerkaserne', and 'Zentralfriedhof'.
- Footer fields:** 'Füge Freunde hinzu', 'Sichtbarkeit und Privatsphäre', and 'Beschreibung der Aktivität'.
- Speichern**: Save button.

Abbildung 8.2: Formular für das Anlegen einer Aktivität innerhalb des Webportals. Es werden generelle Dinge eingegeben, wie der Name der Aktivität oder die Sportart, sowie die Quelle der Strecke.

8 Analyse aus Benutzersicht

zu setzen. Dieser definiert den Startpunkt der Strecke. Sobald ein weiterer Punkt gesetzt wurde, der den ersten Wegpunkt der Strecke definiert, wird eine Route zwischen diesen beiden Punkten bestimmt. Dieser Vorgang wird so lange wiederholt, bis das gewünschte Streckenprofil erstellt wurde. Zusätzlich besteht die Möglichkeit, zwei verschiedene Routenplaner zur Berechnung der Strecken auszuwählen sowie eine einfache Linie zwischen zwei Punkten zu bestimmen.

Das Anbieten zweier Routenplaner war erforderlich, da die unterschiedlichen Routenplaner basierend auf den OpenStreetMaps Daten verschiedene Ergebnisse für dieselben Wegpunkte lieferten. Dies liegt vor allem daran, dass z. B. bei manchen Routenplanern kein Fußgänger über eine »nur für Radfahrer« definierte Straße gehen darf. Da in der Praxis aber z. B. auch Radfahrwege zum Laufen verwendet werden, werden zwei unterschiedliche Routenplaner angeboten. Jedes einzelne Segment zwischen zwei Wegpunkten kann entweder mit einem der beiden Routenplaner bestimmt, oder mit einer Linie direkt verbunden werden. Der Auswahlknopf »Add Marker« erlaubt das Setzen eines weiteren Markers auf der Strecke zwischen zwei Wegpunkten. Mit »Delete Marker« können ausgewählte Wegpunkte wieder gelöscht werden. Abbildung 8.3 zeigt eine fertig erstellte Strecke mit Höhenprofil. Zu erkennen ist, dass bei der Verbindung zwischen Wegpunkt 6 und 7 ein Liniensegment gezeichnet wurde.

Des Weiteren kann die Dauer bestimmt werden, die für das Laufen der Strecke benötigt wurde. Dadurch kann vom System eine Durchschnittsgeschwindigkeit berechnet werden. Zusätzlich wurde auch für die Karte eine Navigationshilfe zu einer bestimmten Adresse oder Stadt integriert.

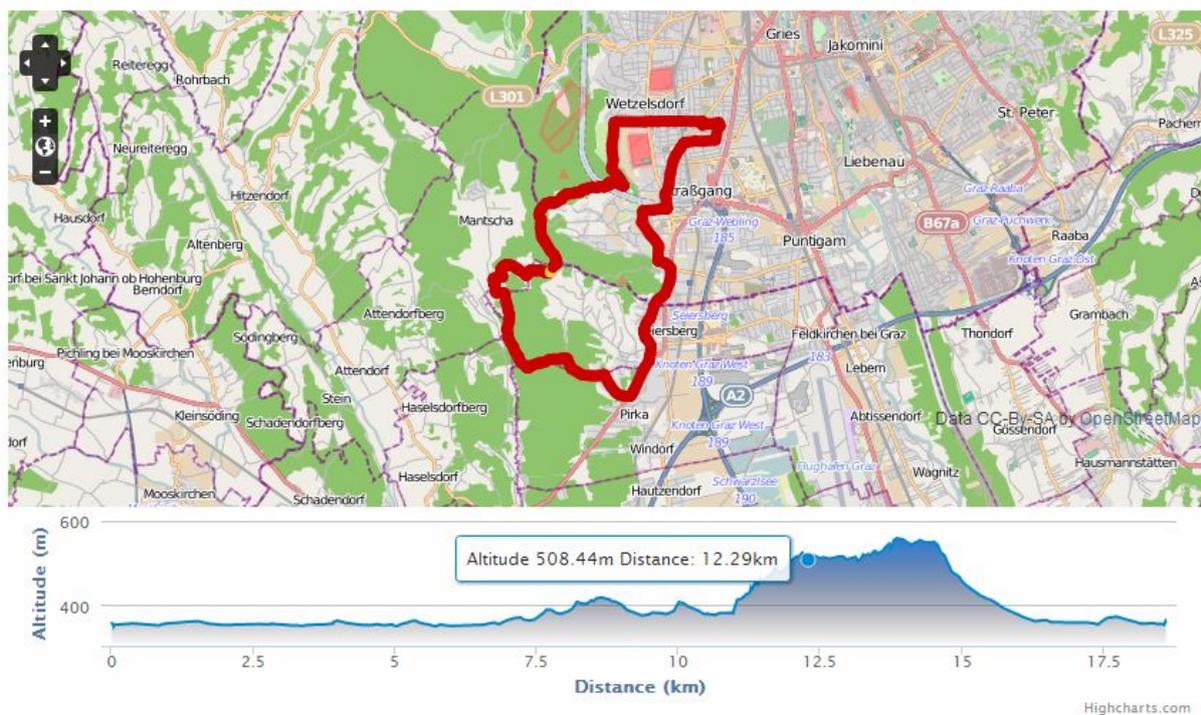


Abbildung 8.4: Nach dem Upload einer GPX-Datei wird die Strecke sofort auf der Karte in Verbindung mit einem Höhenprofil präsentiert.

8.2.2 Ändern einer bestehenden Strecke

Neben dem Routenplaner bestehen zwei Möglichkeiten, bereits bestehende Strecken weiter zu verwenden. Zum Einen können bereits erstellte Strecken aus dem System geladen werden, zum anderen können Streckendaten über eine GPX-Datei hochgeladen werden. Sobald diese Ladevorgänge abgeschlossen sind, wird die Strecke, inklusive Höhendiagramm, präsentiert (siehe Abbildung 8.4). Im nächsten Schritt können diese Routen weiter angepasst werden. Diese Funktionalität hat vor allem dann großen Nutzen, wenn z. B. eine neue Laufstrecke sich nur an wenigen Stellen von einem bereits existierenden Streckenprofil unterscheidet. Damit kann die existierende Strecke rasch auf die neue Route angepasst werden.

Abbildung 8.5 zeigt in drei Schritten die Veränderung der Strecke durch neues Setzen bzw. Verschieben von Wegpunkten. Das erste Bild zeigt eine geladene Strecke ohne Wegpunkte. Durch Auswählen von »Editier« werden in Abständen Wegpunkte auf die Strecke platziert. Ab diesem Zeitpunkt ist fast dieselbe Funktionalität wie beim Routenplaner auch auf diese Strecke anwendbar. Der einzige Unterschied besteht darin, dass kein Wegpunkt willkürlich auf die Karte gesetzt werden kann. Es können nur neue Wegpunkte zwischen zwei Markern hinzugefügt, verschoben und gelöscht werden. Die dritte Grafik in Abbildung 8.5 zeigt das neue Streckenprofil, welches durch Verschieben der Wegpunkte erstellt wurde. Zur Berechnung eines Streckensegments, nachdem ein Wegpunkt verschoben wurde, kann wiederum zwischen zwei Routenplanern bzw. einer einfachen Linie gewählt werden.

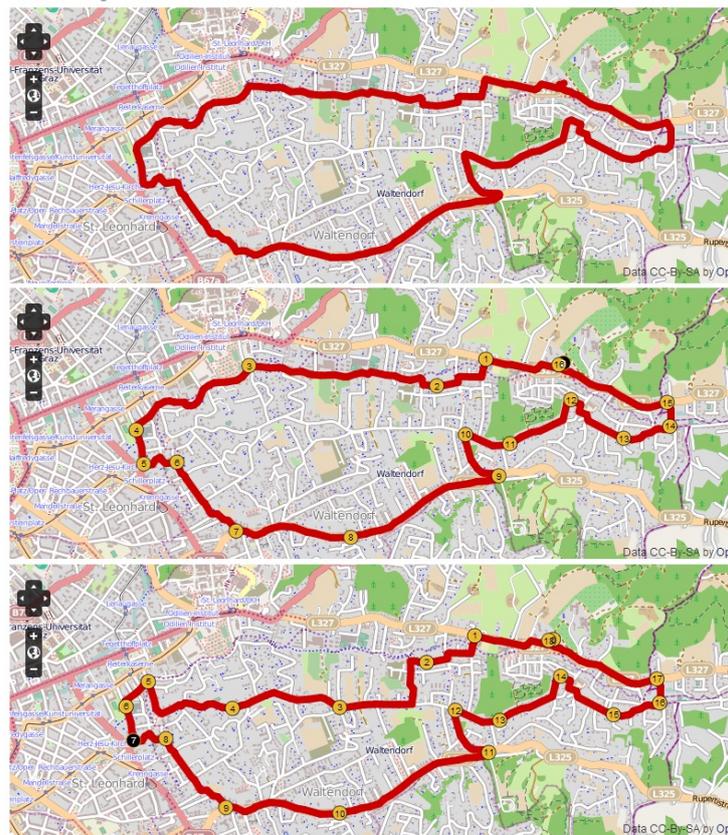


Abbildung 8.5: Durch das Verschieben der Marker ist eine Änderung des Streckenverlaufs möglich.

8.3 Darstellung einer Aktivität

Nach dem Anlegen und Abspeichern zeigt Abbildung 8.6 die fertige Aktivität. Diese besteht aus der Karte mit dem Streckenverlauf, etwaigen gewonnenen Geocaching-Objekten, dem Höhendiagramm sowie einer Tabelle mit allen, die Aktivität betreffenden wichtigen Daten. In diesem Beispiel wurde ein Geocaching-Punkt-Objekt gewonnen, welches als gelbe Fahne in der Karte präsentiert wird.

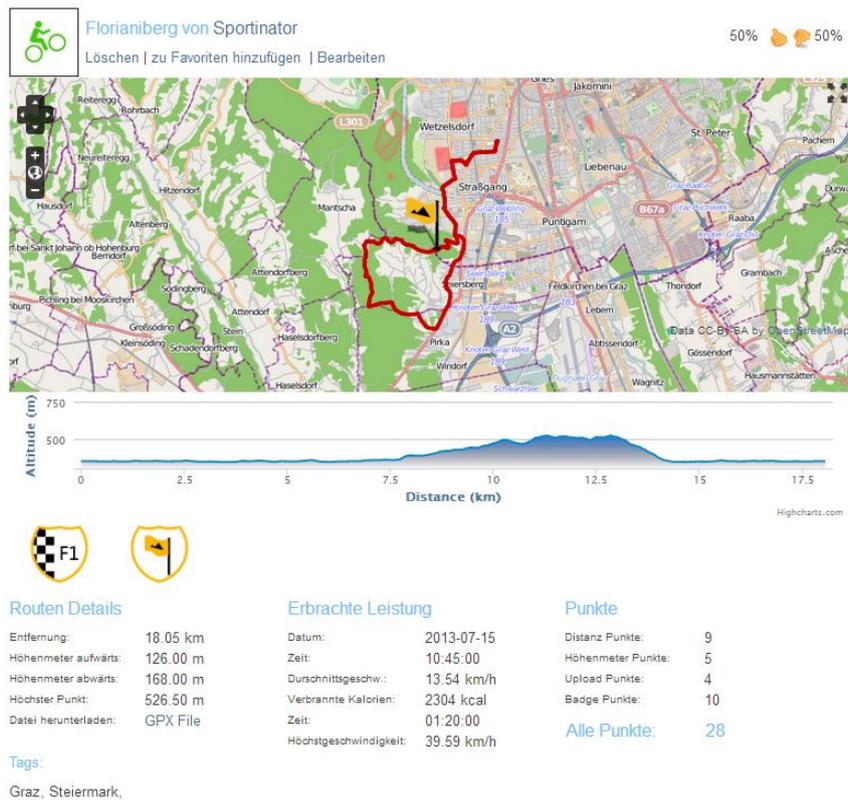


Abbildung 8.6: Ansicht einer Aktivität im Webportal. Stimmt der Streckenverlauf mit einem Geocaching-Objekt, wie in diesem Fall, überein, so wird dieses Objekt auf der Karte mit der Strecke angezeigt.

8.4 Ablauf von Geocaching

Geocaching-Objekte können auf dieselbe Art wie Strecken von Aktivitäten erstellt werden. Eine Auflistung aller Objekte in der Umgebung des Users bietet die Übersichtskarte. Hier kann zwischen Strecken- und Punkt-Objekten bzw. Rennstrecken unterschieden werden. Der User bekommt dadurch Kenntnis über die Position dieser Objekte. Um diese »auf sammeln« zu können, muss z. B. der nächste Lauf bei den Objekten vorbeiführen. Wie bereits beschrieben, wird zunächst eine neue Aktivität angelegt. Die dazugehörige Strecke wird wieder durch Auswahl einer der drei Quellen hinzugefügt. Nach dem Abspeichern der Aktivität wird diese anschließend mit etwaigen Geocaching-Objekten präsentiert.

8 Analyse aus Benutzersicht

Abbildung 8.1 zeigt die Übersichtskarte, auf welcher alle Geocaching-Objekte in der näheren Umgebung von *Graz* markiert wurden. Die Startpunkte virtueller Rennstrecken werden als karierte Fahnen dargestellt. Die gelben Fahnen unterscheiden zum einen Punkt- und Strecken-Objekte, wobei die Punkt-Objekte noch einmal unterschieden werden in Mountainpeak, also Berggipfel, und normale Punkte.

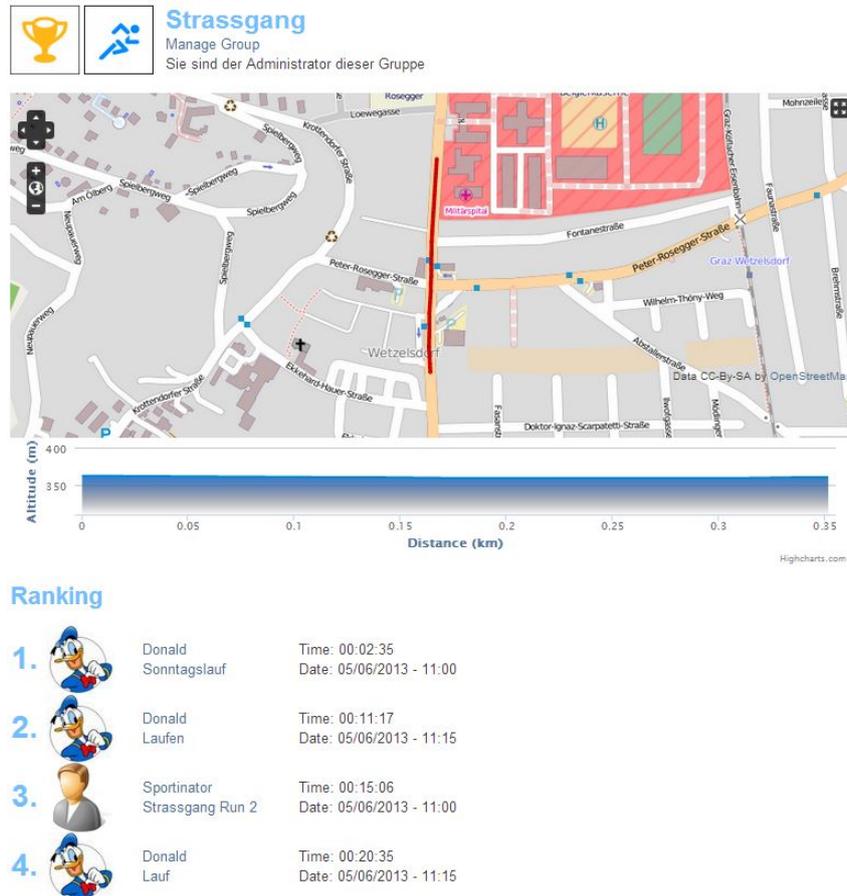


Abbildung 8.7: Ansicht der Rennstrecke in Verbindung mit dem Ranking der einzelnen User.

Virtuelle Rennstrecken

Auf der Präsentationsseite einer Rennstrecke wird zum einen der Streckenverlauf, zum anderen ein Ranking aller Teilnehmer des Wettkampfes dargestellt. Wie Abbildung 8.7 zeigt, besteht die Rangfolge aus dem Namen des Benutzers, die berechnete Zeit die benötigt wurde, sowie einem Link zur Aktivität des Nutzers. Auf der Aktivitäts-Übersichtsseite wird zusätzlich ein Hinweis gegeben, dass diese Aktivität einer virtuellen Rennstrecke hinzugefügt wurde.

8.5 Evaluierung des Webportals

Um neben einer Analyse aus der Benutzersicht auch ein Feedback über die Akzeptanz der Funktionalität des Webportals erhalten zu können, wurde ein Fragebogen ausgearbeitet. Dieser behandelt neben demographischen Grundfragen eine spezielle Benutzeranalyse des implementierten Routenplaners. Die Auswertung des Fragebogens gibt Auskunft über das Handling und die Bedienbarkeit, der in dieser Arbeit erstellten Funktionen. Nachdem auf dem Webportal die *Social-Media* sowie *Gamification* von Sandriesser (2013) durchgeführt wurde, fließen auch diese Punkte in den Fragebogen mit ein. Deren Ergebnisse sollen in dieser Arbeit aber nicht näher erörtert werden.

Der Fragebogen wurde mit der freien Online-Umfrage-Applikation *LimeSurvey*¹ ausgearbeitet. Diese Applikation bietet alle notwendigen Funktionen für die Aufstellung verschiedener Fragebögen und die anschließende exakte Ausgabe der Ergebnisse. Zusätzlich lassen sich mit dem Tool die Antworten und Statistiken in diversen Formaten wie XML oder CSV exportieren und weiterverarbeiten.

8.5.1 Aufbau des Fragebogens

Der erste Teil der Erhebung des Fragebogens bezieht sich auf demographische Gebiete, wie Geschlecht und Alter. Anschließend beginnt bereits eine allgemeine Befragung zur Sportlichkeit des Kandidaten. Hier wird vor allem bestimmt, welche Arten von Sport gemacht werden, wie oft und ob die Aktivitäten allein oder mit Freunden durchgeführt werden. Bevor mit der Befragung des Webportals begonnen werden kann, wird nach alternativen, bereits in Verwendung befindlichen Webportalen gefragt. Durch gezielte Fragen nach den verwendeten Funktionalitäten auf diesen Webportalen kann herausgefunden werden, welche Faktoren die befragten Personen motivieren, diese Portale zu verwenden.

Nach dieser allgemeinen Befragung gehen die nächsten Punkte des Fragebogens gezielt auf das neu erstellte Webportal ein. Dazu wird die befragte Person zunächst aufgefordert, sich ein paar Minuten mit dem Webportal zu beschäftigen, die verschiedenen Menüs auszuprobieren um durch das Portal zu navigieren. Anschließend soll eine neue Aktivität erstellt werden, um den Routenplaner für das Erstellen einer Strecke zu testen. Die Erfahrungen, die dabei gemacht wurden, werden anschließend durch diverse Fragen, welche Performance sowie die Bedienbarkeit des Streckeneditors betreffen, abgefragt.

Abschließend untersucht der Fragebogen die allgemeine Zufriedenheit hinsichtlich dem Webportal. Dazu werden Fragen über die Menüführung, das Design sowie die Geschwindigkeit des Portals gestellt. Des Weiteren soll angegeben werden, welche Aspekte besonders gut gefallen haben, und welche Dinge auf der Webseite noch verbessert werden müssen. Wie im gesamten Fragebogen wird hier u. a. mit offene Antworten gearbeitet, sowie wird eine Skalierung zwischen 1 (trifft zu) bis 5 (trifft nicht zu) angeboten. Gerade durch die Möglichkeit, offene Antworten geben zu können, wird versucht, möglichst viele unterschiedliche Meinungen und Ansichten zum Webportal zu erhalten. Die Auswertung und Umsetzung der Antworten soll in Zukunft zu einer Verbesserung der Akzeptanz und Funktionalität des Webportals führen.

¹<http://www.limesurvey.org/index.php/de/> – zuletzt abgerufen am 13.08.2013.

8.5.2 Stakeholder Gruppe

Es wurde versucht, gezielt jene Personen den Fragebogen ausfüllen zu lassen, welche zumindest einmal in der Woche aktiv Outdoor-Sport betreiben. Dies führte daher, dass eine gewissen Sportaffinität und die Erfahrung, die dabei einhergeht, eine bessere Beantwortung der Fragen ermöglicht. Des Weiteren wurde versucht, Teilnehmer zu finden, welche bereits aktiv mit Webportalen oder Applikationen für Smartphones ihre sportlichen Leistungen aufzeichnen und analysieren. Auch hier sollen wieder die Erfahrungen der Teilnehmer mit solchen Monitoring-Systemen in die Entwicklung des neu implementierten Webportals einfließen.

8.5.3 Auswertung

Im folgenden Abschnitt wird eine kurze Auswertung und Interpretation der Antworten, welche im Fragebogen gegeben wurden, beschrieben. Unterteilt wird dieser Abschnitt in die Resultate der demographischen Fragen mit dem allgemeinen Sportverhalten der Kandidaten, und den Antworten, welche speziell auf das neue Webportal abzielen. Die Umfrage wurde 31 mal beantwortet, jedoch können 3 Bögen als ungültig angesehen werden da sie nicht vollständig ausgefüllt oder zu Ende gebracht wurden, woraus folgt, dass 28 gültige Ergebnisse vorliegen.

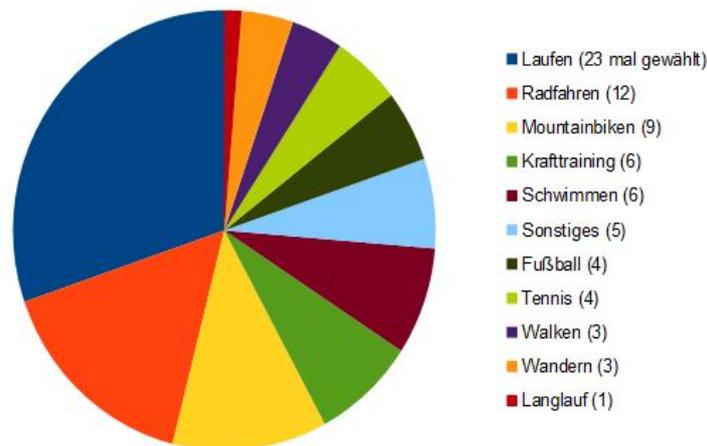


Abbildung 8.8: Dieses Tortendiagramm illustriert, mit welcher Häufigkeit die verschiedenen Sportarten gewählt wurden.

Demographische Fragen und allgemeines Sportverhalten

An der Umfrage nahmen 20 männliche und 8 weibliche Personen teil. Der Großteil der befragten Personen waren Studenten (11 Personen, 39,29% der Antworten) bzw. Akademiker (10 Personen, 35,71%) gefolgt von Personen mit einem Lehramtsabschluss (4 Personen, 14,29%) und einem Maturanten (3,57%). Die restlichen zwei Personen gaben unter »Sonstiges« Angestellter

8 Analyse aus Benutzersicht

und Beamter an. 18 Personen waren im Alter zwischen 26-35, vier zwischen 15-25 und jeweils drei zwischen 36-45 bzw. älter als 46 Jahre.

14 Personen gaben an, wöchentlich drei bis vier mal Sport zu treiben, gefolgt von neun Antworten bei zwei mal und fünf Kandidaten betreiben Sport nur einmal in der Woche. Aus diesen Ergebnissen lässt sich auf einen sportlichen Kandidatenkreis schließen. Abbildung 8.8 zeigt die grafische Auswertung der Angaben nach den getätigten Sportarten. Die beliebteste Sportart, welche 23 mal und damit am häufigsten gewählt wurde, war Laufen, gefolgt von Radfahren (12) und Mountainbiken (9). Bei der Frage, ob Sport alleine oder zusammen mit Freunden praktiziert wird, wurde ersteres mit 25 mal am öftesten angegeben, gefolgt von »zusammen mit Freunden« mit 14 Angaben. Drei bzw. zwei mal wurde »im Verein« bzw. »mit der Familie« angegeben. Dies könnte daher führen, dass es nur selten gelingt, ein Training in der Gruppe zu organisieren und daher das Einzeltraining regelmäßiger durchgeführt wird. Bei diesen letzten beiden Fragen war eine mehrmalige Antwort für die Kandidaten möglich.

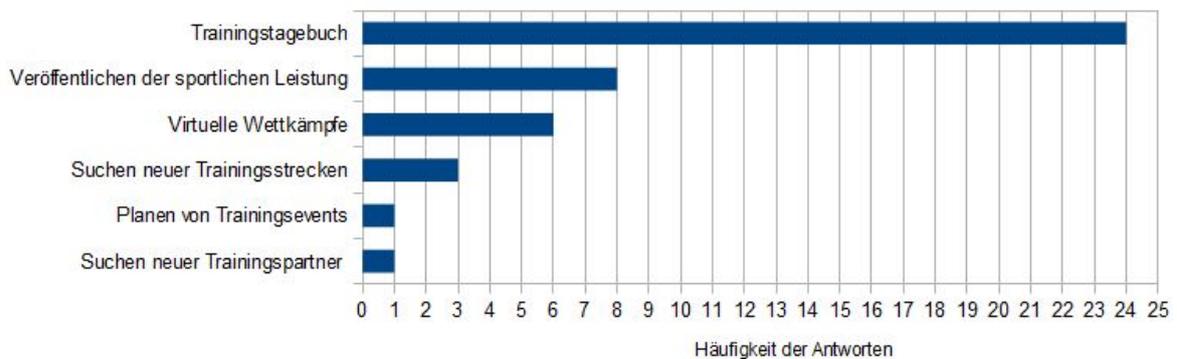


Abbildung 8.9: Statistik der Häufigkeit der Antworten auf die Frage, welche Funktionen bei Software zur Kontrolle von sportlichen Leistungen verwendet werden.

21 Teilnehmer haben angegeben, manchmal, vier nie und drei immer Software zur Unterstützung ihres Trainings zu verwenden. Auf die Frage, welche Software oder Portale zur Leistungsüberprüfung verwendet werden, wurden 24 Antworten gegeben und vier mal keine Angabe gemacht. Diese setzen sich zusammen aus 13 Angaben für *Runtastic*, sechs für *Endomondo*, drei für *Garmin Connect*, und jeweils eine Antwort für *JAFF Military Workout Trainer* und *SportsTracker*. Abbildung 8.9 zeigt, welche Funktionen bei der Software von den Kandidaten am häufigsten Verwendung finden. Der Spitzenwert bei dieser Frage, die eine Mehrfachauswahl zugelassen hat, war die Verwendung des Trainingstagebuchs mit 24 Antworten. Mit genau 1/3 des Topwerts belegt die Antwort »Veröffentlichen der sportlichen Leistung« den zweiten Platz. Die weniger verwendeten Funktionen sind die Suche nach Trainingsstrecken bzw. Trainingspartner sowie das Planen von Trainingsevents.

Durch die Analyse und Darstellung der sportlichen Leistungen werden die Kandidaten von der Software indirekt zu mehr Sport motiviert. Abbildung 8.10 zeigt die Antwortstatistik auf die Frage, welche Funktionen von Trainingssoftware sich positiv auf die Motivation auswirken. Auch diese Frage gab die Möglichkeit einer Mehrfachantwort. Die beiden Topwerte mit 25 bzw. 17 Antworten waren die Analyse der sportlichen Leistungen sowie das Trainingstagebuch mit dem Leistungsverlauf. Ähnlich der Statistik in Abbildung 8.9 liegt die Möglichkeit

8 Analyse aus Benutzersicht

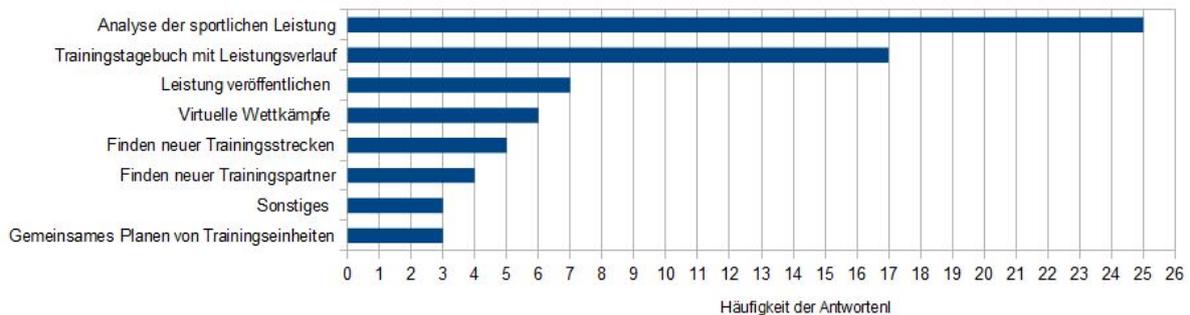


Abbildung 8.10: Statistik der Häufigkeit der Antworten auf die Frage, welche Funktionen bei Software zur Kontrolle von sportlichen Leistungen sich motivierend auf Leistungen auswirken.

zum Auffinden neuer Trainingsstrecken oder Trainingspartner im unteren Bereich. Diese Auswertung lässt darauf schließen, dass die Kandidaten die Software primär dazu verwenden, die eigene Trainingsleistung aufzuzeichnen und zu analysieren. Nur teilweise finden die anderen Möglichkeiten, welche geboten werden, wie das Auffinden neuer Trainingsstrecken, Verwendung.

15 der 28 befragten Personen halten die persönliche Trainingsleistung privat und teilen diese nicht mit der Community der verwendeten Software. Elf teilen diese manchmal und zwei immer. Daraus lässt sich schließen, dass die Software primär zur Analyse und Überwachung verwendet wird, und weniger dafür, mit anderen Sportlern im sozialen Netzwerk zu interagieren.

Fragen zum Webportal und dessen Funktionalität

Die allgemeinen Fragen zum Design, Navigation und Funktionalität des Webportals wurden mehrheitlich positiv bewertet und wurden in der Arbeit von Sandriesser (2013) ausführlich ausgewertet und diskutiert. Da für diese Arbeit vor allem die Evaluierung des Routenplaners von Interesse ist, wird auf die Auswertung der dazu passenden Fragen näher eingegangen.

Der Großteil der Kandidaten sieht in dem angebotenen Streckeneditor eine Alternative zur Verwendung von GPS-Geräten. Bei der Frage »Würden sie den Routenplaner verwenden« wurden 20 Angaben bei der Antwort »Gelegentlich (Wenn ich kein GPS Gerät dabei habe)« getätigt. Durch die Möglichkeit einer Mehrfachantwort kam es zu 8 Antworten bei »Immer« und drei Antworten bei »Nie (Ich verwende ein GPS Gerät)«. Die Antwort »Nie (Ich zeichne keine Strecken auf)« wurde kein einziges mal ausgewählt. Abbildung 8.11 zeigt zusammenfassend die Auswertung der drei Fragen „Ich war in der Lage die gewünschte Route zu erstellen (blau)“, „Das Handling des Routenplaners war kompliziert (rot)“ und „Die Geschwindigkeit des Routenerstellens war in Ordnung (grün)“. Die Y-Achse beschreibt die Häufigkeiten der auftretenden Antworten, wobei die X-Achse von „1(trifft zu)“ bis „5(trifft nicht zu)“ reicht. Durch dieses Diagramm wird ersichtlich, dass zwar die gewünschte Route erstellt werden konnte (arithmetisches Mittel 1,54), das Handling und die Geschwindigkeit des Routenplaners überzeugte, jedoch nicht alle Kandidaten. Bei der Frage nach dem Handling lag das arithmetische Mittel bei 3,07 und bei der Geschwindigkeit bei 2,82.

8 Analyse aus Benutzersicht

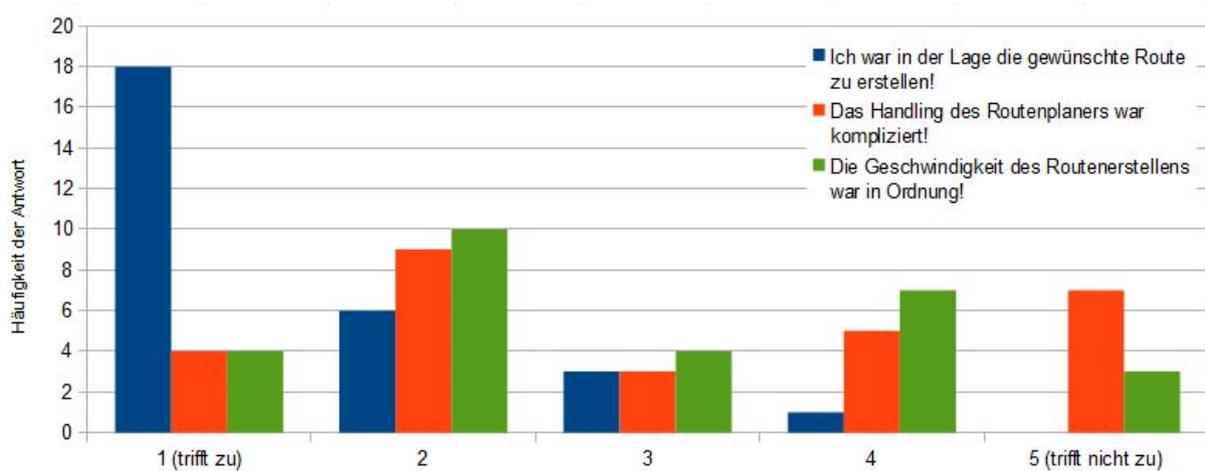


Abbildung 8.11: Zusammenfassung der drei Fragen bezüglich Funktion und Performance des Routenplaners. Die X-Achse zeigt die Zufriedenheit der befragten Personen von 1 (trifft zu) bis 5 (trifft nicht zu), die Y-Achse spiegelt die Häufigkeit der gegebenen Antworten wieder.

Am Ende der Umfrage wurden verschiedene offene Fragen gestellt. Eine davon war, was besonders gut auf dem Webportal gefallen würde. Von den 28 Antworten, die hier gegeben wurden, wurde der im Zuge dieser Arbeit implementierte Routenplaner 14 mal positiv erwähnt. Hervorzuheben sind hier vor allem die Aussagen „Einfaches Bearbeiten/Ändern von bereits gelaufenen Routen“, „Wiederverwenden von Strecken“ oder „Vorhandensein alle Waldwege/Feldwege im Routenplaner“. Diese Angaben zeigen, dass die neuen Funktionalitäten des im Zuge dieser Arbeit entwickelten Webportals bei den befragten Personen Beachtung fanden. Bereits in der Evaluierung aktueller Anwendung wurde herausgefunden, dass das Wiederverwenden und Ändern von Strecken ein wichtiger Punkt bei einer neuen Anwendung sein könnte. Ein weiterer Punkt, der in dieser Arbeit aufgezeigt wurde, ist das dichtere Streckennetz von *OpenStreetMaps* Karten im Vergleich zu denen von *Google Maps* oder *Bing Maps*. Deshalb wurde bei der Implementierung viel Wert darauf gelegt, mit den Karten von *OpenStreetMaps* zu arbeiten. Diese Befragung zeigt, dass die implementierten Funktionalitäten bei der Zielgruppe des Webportals positives Feedback hinterlassen. Weitere fünf Angaben bei den Dingen, die den befragten Personen besonders gut gefallen haben, bezogen sich auf die implementierten Geobadges sowie virtuelle Rennstrecken und Wettkämpfe. Eine eher geringere Aufmerksamkeit für diese Punkte könnte daher kommen, dass es ein längeres Arbeiten mit dem Portal benötigt, um alle Funktionen und Möglichkeiten zu erkennen bzw. zu verwenden. Die restlichen offenen Antworten zu dieser Frage geben an, dass den befragten Personen das „Design“, „die gesamte Idee“ oder auch der „Social Network Charakter“ sehr gut gefallen haben.

Die Frage nach Dingen, die auf dem Portal verbessert werden könnten, wurden vor allem mit „Performance und Ladezeiten“, „Routenplaner noch einfacher gestalten“ sowie mit „Wettkämpfe besser erklären“ beantwortet. Generell wurde das Fehlen von Hilfestellungen für die Funktionen bzw. von Erklärungen bemängelt. Die Ladezeit und Performance des Routenplaners ist insofern ein Problem, da die Streckendaten von einem externen Server angefragt werden müssen, da für dieses Projekt kein eigener Routenplaner implementiert wurde. Durch das Feedback der mangelhaften Bedienbarkeit und Erklärung des Routenplaners bzw. der Wettkämpfe wurde von den befragten Personen Denkanstöße für zukünftige Arbeiten am

Portal gegeben. Diese könnten dahin gehen, mehr Informationszeichen mit Tooltips als Erklärungen für die Funktionen hinzuzufügen. Dadurch wird es möglich sein, den Benutzer besser und gezielter die Bedienung zu erklären. Die notwendigen Schritte, welche auf dem Webportal unternommen werden müssen, um die negativen Aspekte, welche in der Benutzerumfrage herausgekommen sind, zu verbessern, werden detaillierter im folgenden Kapitel diskutiert.

8.6 Zusammenfassung

Diese Usage Analyse zeigte den Ablauf der wichtigsten Funktionen, die im Zuge dieser Masterarbeit implementiert wurden. Hauptbestandteil war die Möglichkeit, innerhalb des Anlegens einer sportlichen Aktivität Strecken editieren zu können. Es wurde gezeigt, wie neue Strecken aufgebaut werden und wie bestehende Strecken durch Verschieben oder Löschen von Wegpunkten editiert werden können. Mit der Übersichtskarte kann die nähere Umgebung nach Geocaching-Objekten untersucht werden. Mittels jener Aktivitäten, deren Streckenverlauf an solchen Objekten vorbeiführt, werden diese eingesammelt.

Bei der Präsentation der Aktivität werden anschließend etwaige Objekte eingetragen. Zusätzlich werden hier auch alle wichtigen Informationen die Strecke und Aktivität betreffen, ausgegeben. Mit Hilfe eines Fragebogens konnte eine objektive Analyse des Webportals erstellt werden. Die Ergebnisse zeigen, dass der Aufbau, das Design sowie die Funktionalität sehr positiv bewertet wurden. Zwar wurde das Handling des Routenplaners als zu schwierig angegeben, jedoch war es den Kandidaten dennoch möglich den gewünschten Streckenverlauf zu erstellen. Speziell der Routenplaner wurde als eine sehr wichtige Funktionalität des Webportals hervorgehoben.

Dieser Abschnitt stellt das Ende der Umsetzung der Geodatenverarbeitung in einer Social-Media Umgebung dar. Die folgenden Kapitel werden sich mit den Erfahrungen, die in der Ausarbeitung und Umsetzung dieses Themas gemacht wurden sowie mit dem Ausblick über zukünftige Arbeiten befassen.

9 Lessons Learned

In diesem Kapitel werden Erfahrungen beschrieben, die im Zuge der Umsetzung dieses Projektes gemacht wurden. Der theoretische Teil beschreibt vor allem Open Services und deren Zuverlässigkeit und Genauigkeit im Vergleich zu proprietären Services großer Anbieter. Der praktische Teil wird sich mit den Herausforderungen im Zuge der Implementierung befassen. Hier wird vor allem auf das Zusammenspiel von Drupal und *OpenLayers* eingegangen. Des Weiteren beschreibt dieser Abschnitt die Schwächen der *OpenLayers* Bibliothek.

9.1 Theoretischer Teil

Bei der Evaluierung bereits verwendeter Sportportale wurde schnell ersichtlich, dass eine sehr große Auswahl guter Webseiten für die Auswertung sportlicher Aktivitäten vorhanden sind. Nur zusätzliche bzw. die Verbesserung aktueller Funktionalität würde ein Abheben von bestehenden Portalen ermöglichen. Ein daraus resultierender Punkt war der Einsatz eines Routenplaners basierend auf der OpenStreetMap Karte. Nachdem die Implementierung des Routen-Berechnungsalgorithmus im Zuge dieser Arbeit nicht durchgeführt wurde, musste auf externe und frei zugängliche Planer zurückgegriffen werden. Hier entstand das Problem, dass jeder dieser Routenplaner diverse Strecken oder Kreuzungen anders interpretiert. So ist es bei manchen Routenplanern nicht möglich gewesen, z. B. als Fußgänger auf einem Weg zu gehen, welcher als Fahrradweg deklariert wurde. Bei anderen war es wiederum nicht möglich, auf Waldwegen mit dem Fahrrad zu fahren, was vor allem beim Anlegen von Mountainbiketouren zu Problemen führte.

Ein Problem, das sowohl bei den Routenplanern als auch bei den anderen freien Geodiensten auftrat, war die Zuverlässigkeit der Server. Sehr oft waren die Dienste nicht verfügbar, und die Anfrage via API verlief in einem Time-Out. Auf einer Social-Community Webseite, in der diese freien Geodienste gebraucht werden, führt dies zu schwerwiegenden Problemen. Dies war ein weiterer Grund, zwei Routenplaner bei der Erstellung einer Strecke anzubieten, um ein Backup-System zu haben, falls einer der beiden ausfallen würde. Hier ist klar die Verwendung proprietärer Dienste großer Anbieter wie Google oder Bing zu bevorzugen, da diese mit Sicherheit eine größere Zuverlässigkeit besitzen wie freie Dienste.

9.2 Praktische Umsetzung

Die Implementierung kann in zwei Bereiche geteilt werden: Zum einen die alleinige Implementierung in *OpenLayers*, zum anderen die Verwendung des *OpenLayers*-Moduls in Drupal. Dies erforderte sehr oft interdisziplinäres Denken zwischen den einzelnen Programmiersprachen, da in JavaScript, PHP und MySQL programmiert wurde und die gesamte Funktionalität

von Drupal im *Back-End* bekannt sein musste. Trotz seines großen Funktionsumfangs wurden auch bei *OpenLayers* während der Implementierungsphase Schwächen festgestellt. Diese beziehen sich vor allem auf Darstellungen und Aussehen. So ist es mit *OpenLayers* nicht möglich, verschiedene Effekte auf das Erscheinen oder Verschwinden von Markern oder Layern anzuwenden.

Des Weiteren fehlt *OpenLayers* in seiner aktuellen Version 2.0 eine gewisse Dynamik beim Zoomen bzw. Bewegen der Karte. Das Fehlen dieser Effekte lässt die gesamte Karte statisch und schwerfällig wirken. Neben der Präsentation von Streckeninformationen auf einer Karte, mussten »im Hintergrund« exakte Berechnungen der Streckendaten durchgeführt werden. Dazu mussten trigonometrische Grundlagen erlernt und umgesetzt werden. Speziell bei der Abstandsberechnung zwischen geografischen Objekten wurde ein bis dahin für den Autor nicht bekannter *Satz des Heron* verwendet.

Zusätzlich zu den Funktionalitäten, welche mit *OpenLayers* umgesetzt wurden, musste auch das CMS *Drupal* verstanden werden. Speziell bei der Umsetzung von Geocaching war es nötig, den gesamten Ablauf des *Drupal*-Systems zu verstehen, da es dafür nicht reichte, nur im *Back-End* zu arbeiten.

9.3 Interpretation der Benutzerbefragung

Die Benutzerbefragung ergab, dass es den Kandidaten zwar möglich war, eine Strecke mit Hilfe des Routenplaners anzulegen, sie diesen Vorgang aber als zu kompliziert und den Planer als zu langsam empfanden. Im Zuge der Implementierung wurde versucht, die Funktionalität so einfach und intuitiv wie möglich zu gestalten. Das Erstellen einer Route ist vom Prinzip ein sehr komplexer Vorgang, da durch die automatische Berechnung der Strecke zwischen zwei Punkten *A* und *B* möglicherweise nicht die Strecke ausgegeben wird, die sich der User erwartet hat. Das zusätzlich Hinzufügen von Punkten zwischen *A* und *B* wird womöglich von den Benutzern als kompliziert empfunden, ist aber notwendig um den gewünschten Verlauf der Route zu erreichen.

Nachdem nur die Darstellung und nicht auch die Berechnung der Strecke auf dem Webportal stattfindet, benötigt die Anfrage nach dem Verlauf der Route eine gewisse Zeit. Diese Dauer ist auch von der Geschwindigkeit des Servers abhängig, welcher die Streckeninformationen hält. Dieses Design des Portals birgt neben den Problemen mit der Geschwindigkeit auch anderes Gefahrenpotenzial: Sollte der Routen-Server ausfallen, die Berechnung zur Gänze eingestellt werden oder die API geändert bzw. beschränkt werden, so wird auch auf dem Webportal die Routenberechnung deaktiviert. Deshalb wurden von Anfang an zwei verschiedene Berechnungsdienste angeboten. Trotzdem sollte eine intern liegende Routenberechnung überdacht und für zukünftige Arbeiten vorgesehen werden.

10 Zusammenfassung und Ausblick

Der Einsatz von GPS-Techniken zur Positionsbestimmung oder Navigation ist durch die Verwendung in Smartphones bzw. die billiger werdende Technik für beinahe jedermann erreichbar. Dadurch befassen sich viele Anwendungen damit, durch Applikationen und Webseiten die GPS Techniken in Sport- und Freizeitaktivitäten einfließen zu lassen. Diese Masterarbeit zeigt die Erstellung eines neuen Webportals, welches durch die Verbindung und Verbesserung vorhandener Ideen einen innovativen Einsatz dieser Technik versucht.

Um eine neue Plattform zu entwickeln, müssen zuerst die Grundlagen recherchiert und verstanden werden. Deswegen begann dieses Projekt mit der Analyse von Geoinformationssystemen und im speziellen deren Einsatz im World Wide Web. Dies führte zu den Standardisierungen von Schnittstellen dieser Daten. Dadurch wird ein Austausch umfassender Geodaten zwischen verschiedenen Diensten erst möglich gemacht. Diese Schnittstellen erlauben es, auf Datenbanken und Berechnungen von Geoinformationen zuzugreifen, wodurch z. B. die Entwicklung eines Routenplaners auf dem neuen Portal effizient möglich wurde.

Das neu entwickelte Portal vereint Ideen von Sportwebseiten mit jenem des Geocachings. Dazu wurden zunächst vorhanden Applikationen und Webseiten, welche diese Themen befassen, analysiert. Daraus konnten die wichtigsten Punkte, wie die Präsentation von Strecken oder der Einsatz eines Routenplaners, erkannt werden. Neben diesen Analysen für Outdoorstrecken beinhaltet das Portal auch Geocaching Elemente. Dadurch können mit Aktivitäten verschiedene virtuelle Objekte eingesammelt werden. Diese können auf einen Punkt konzentriert, längere Strecken oder ganze virtuelle Rennstrecken sein. Um diese Punkte umsetzen zu können, musste vor der Implementierung die richtigen Tools und Frameworks gefunden werden. Die Entscheidung fiel zum einen für die Basis der Implementierung auf das Content-Management-System Drupal 7, zum anderen für die Umsetzung der Geodatenverarbeitung auf OpenLayers.

Der Implementierungsprozess zeigte, in welche verschiedenen Bereiche das Arbeiten mit Geodaten führt. So mussten auch mathematische Grundlagen von Geometrie und im speziellen Trigonometrie umgesetzt werden. Dies wurde gebraucht u. a. bei der Berechnung von Distanzwerten der Route sowie bei der Berechnung von Entfernungen der Geocaching-Objekten zu den Strecken. Des Weiteren zeigt die Implementierung die Zusammenarbeit verschiedener Frameworks. So mussten verschieden Daten des CMS-Drupal in OpenLayers präsentiert werden. Zusammenfassend wurde die Umsetzung der Anforderungen durch eine Analyse aus der Benutzersicht präsentiert. Diese illustriert alle wichtigen Punkte der implementierten Webumgebung.

Nachdem die Grundfunktionalität der Geodatenverarbeitung innerhalb von Drupal im Zuge dieser Masterarbeit durchgeführt wurde, können aus den Auswertungen des Lessons-Learned Kapitel sowie der Analyse aus Benutzersicht bereits Punkte gefunden werden, welche zukünftig das Webportal verbessern und erweitern könnten. Die wichtigsten Punkte werden

in dem folgenden Abschnitt diskutiert. Zunächst werden die Vorteile diskutiert, die ein intern liegender Routenplaner mit sich bringen würde. Des Weiteren wird auf die Möglichkeiten der Koppelung des Webportals mit einer mobilen App eingegangen. Abschließend wird auf die Grafik und das Design der Datenaufbereitung und des Webportals hingewiesen.

Interner Routenplaner

Obwohl bei der Implementation der Anbindung eines externen Routenplaners darauf geachtet wurde, die versendeten Datenpakete so klein wie möglich zu halten, benötigt die Anfrage je nach Geschwindigkeit des Servers und der Übertragung eine gewisse Zeit. Dadurch kann sich die Erstellung einer Route für den Benutzer als mühsam erweisen und er könnte von der Verwendung absehen. Dieses Problem kann nur durch einen Routenplaner gelöst werden, der lokal am Server arbeitet. Deshalb wäre für die Zukunft die Implementierung eines solchen für dieses Portal zu überlegen. Dieser selbst implementierte Routenplaner könnte dann auch auf die gewünschten Spezifizierungen besser eingestellt werden. Damit wäre das Webportal - zumindest was das Routenservice betrifft - unabhängig vom externen Server.

Mobile Applikation

Auf Dauer wird es nötig sein, zusätzlich zum Webportal eine mobile Applikation für alle gängigen mobilen Betriebssysteme zu entwickeln. Mit dieser App sollte es möglich sein, die Routen direkt aufzeichnen zu können, eine Auswertung zu sehen, sowie die Informationen an das Webportal zu senden. Da Geocaching eine große Rolle spielt, sollte die App diese Objekte direkt anzeigen. Somit könnten auch Start- und Endpunkte von virtuellen Rennstrecken leichter gefunden werden.

Grafik und Design

Natürlich ist eine grafische Präsentation immer subjektiv zu betrachten, ob sie gefällt oder nicht, jedoch wäre die Aufbereitung und die Usability der einzelnen Funktionen noch zu überdenken. Speziell bei dem Zusammenspiel vieler verschiedenen Grafiken und Farben sollte ein ausgebildeter Webdesigner ein optimales Design finden können. Social-Community Webportale bestehen hauptsächlich aus Inhalten der Benutzer, welche so akkurat und einfach dargestellt werden müssen, dass die User der Seite schnell die wichtigsten Dinge erkennen können, ohne lange danach suchen zu müssen.

Anhang A

Literatur

- Adam, M. (2012). Javascript implementation of the douglas peucker path simplification algorithm. Zugriff 15. Mai 2013, unter <https://gist.github.com/adammiller/826148>
- Aetrex Worldwide, I. (2013). Aetrex gps shoes. Zugriff 30. Mai 2013, unter <http://www.aetrex.com/aetrex-gps-shoe/?cat>
- Bajaj, R., Ranaweera, S. L. & Agrawal, D. P. (2002). Gps: location-tracking technology. *Computer*, 35, 92–94.
- Boulos, M. K. & Honda, K. (2006). Web gis in practice iv: publishing your health maps and connecting to remote wms sources using the open source umn mapserver and dm solutions maplab. *International Journal of Health Geographics*, 5, 1–7. doi:10.1186/1476-072X-5-6
- Butcher, M., Garfield, L. & Wilkins, J. (2010). *Drupal 7 module development*. Packt Publishing, Limited.
- Byron, A., Berry, A. & De Bondt, B. (2012). *Using drupal*. O'Reilly Media.
- CartouCHE. (2012). Cartography for swiss higher education. Zugriff 27. Mai 2013, unter http://www.e-cartouche.ch/content/reg/cartouche/webservice/en/html/webservice_LSummary.html
- Cham. (2013). Raumbezogene Daten. Zugriff 15. Mai 2013, unter <http://www.landkreis-cham.de/IKGIS/AllgemeineInformationen/GeoDaten.aspx>
- Chen, H. (2007). Geospatial semantic web. *An invited talk at UMBC. CMSC*.
- Cox, S., Cuthbert, A., Lake, R. & Martell, R. (2001). Geography markup language (gml) 2.0. URL: <http://www.opengis.net/gml/01-029/GML2.html>.
- Dalenberg, A. (2013). Ingress, google's underground game, is being played all around you. Zugriff 31. Mai 2013, unter <http://upstart.bizjournals.com/news/technology/2013/05/24/ingress-the-game-taking-over-the-world.html?page=all>
- de La Beaujardière, J. (2012). *Web map service implementation specification*. Open GIS Consortium Inc.
- Drupal. (2013). Come for the software, stay for the community. Zugriff 21. Juni 2013, unter <https://drupal.org/>
- Drupalfacts. (2012). All you ever wanted to know about Drupal. Zugriff 15. Mai 2013, unter <http://drupalfacts.mogdesign.eu/>
- drupal.org. (2012). Mailing lists. Zugriff 15. Mai 2013, unter <http://drupal.org/ mailing-lists>
- Duden. (2013). Duden- crowdsourcing. Zugriff 10. April 2013, unter <http://www.duden.de/rechtschreibung/Crowdsourcing>

Literatur

- Endomondo. (2013). Endomondo — Community based on free GPS tracking of sports. Zugriff 9. März 2013, unter <http://www.endomondo.org>
- Ernesto Rodríguez, C. S. M. & Belz, J. E. (2006). A global assessment of the srtm performance. *Photogrammetric Engineering and Remote Sensing*, 72, 249–260.
- Fraguada, L. (2013). Comparing elevation data. Zugriff 15. April 2013, unter <http://www.grasshopper3d.com/group/ghowl/forum/topics/comparing-elevation-data>
- Gabler, S. (2013). tagstore — Project home page. Zugriff 10. April 2013, unter <http://www.wirtschaftslexikon.gabler.de/Definition/outsourcing.html>
- Garmin International, I. (2013). Opencaching. Zugriff 3. Juni 2013, unter <http://www.opencaching.com/>
- Genesys. (2013). Das europäische Navigationssystem Galileo. Zugriff 10. April 2013, unter <http://www.genesys2010.eu/technik/galileo-was-ist-der-sinn-eines-europaischen-gps-systems/>
- GeoNames. (2013). Geonames fulltextsearch. Zugriff 5. April 2013, unter <http://www.geonames.org/search.html?q=Perth&country=>
- Gesch, B., Muller, J. & Farr, T. G. (2006). The shuttle radar topography mission-data validation and applications. *Photogrammetric Engineering and Remote Sensing*, 72, 233.
- GitHub. (2012). Openlayers switcher module. Zugriff 13. Mai 2013, unter https://github.com/LambethCouncil/opencouncil/blob/master/sites/all/modules/custom/openlayers_switcher/openlayers_switcher.module
- GmbH, R. (2013). Runtastic- makes sports funtastic. Zugriff 26. April 2013, unter <http://www.runtastic.com>
- Goodchild, M. (2007). Citizens as sensors: the world of volunteered geography. *GeoJournal*, 69, 211–221. doi:10.1007/s10708-007-9111-y
- Google. (2012a). Google maps api-web services – google developers. Zugriff 26. April 2013, unter <https://developers.google.com/maps/documentation/webservices/>
- Google. (2012b). Google maps/google earth apis terms of service. Zugriff 26. April 2013, unter https://developers.google.com/maps/terms?hl=de#section_10_12
- GooglePlay. (2013). SummitLynx - Gipfelbuch. Zugriff 31. Mai 2013, unter <https://play.google.com/store/apps/details?id=at.bluesource.summitlynx.activities&hl=de>
- Graf, H. (2008). *Drupal 6: Websites entwickeln und verwalten mit dem Open Source-CMS*. Open Source Library. Addison Wesley Verlag. Zugriff unter <http://books.google.ch/books?id=wSphDIRiKBwC>
- Groundspeak, I. (2013). Geocaching. Zugriff 3. Juni 2013, unter <http://www.geocaching.com/>
- Guo, D., Wu, K., Zhang, Z. & Xiang, W. (2012). Wms-based flow mapping services. In *Services (services), 2012 ieee eighth world congress on* (S. 234–241). doi:10.1109/SERVICES.2012.37
- Haklay, M. & Weber, P. (2008). Openstreetmap: user-generated street maps. *Pervasive Computing, IEEE*, 7, 12–18. doi:10.1109/MPRV.2008.80

- helpster.de. (2013). GPS-Sender für Katzen - so klappt die elektronische Suche. Zugriff 30. Mai 2013, unter <http://www.helpster.de/gps-sender-fuer-katzen-so-klappt-die-elektronische-suche.112528>
- Highcharts. (2013). Free - Non-commercial. Zugriff 31. Mai 2013, unter <http://shop.highsoft.com/highcharts.html>
- Hill, D. J. (2013). A google surprise — worldwide alternative reality game ingress revealed. Zugriff 31. Mai 2013, unter <http://singularityhub.com/2012/11/17/a-google-surprise-worldwide-alternative-reality-game-ingress-revealed/>
- Jansen, M. & Adams, T. (2010). *OpenLayers: Webentwicklung mit dynamischen Karten und Geodaten*. professional reference. Open Source Press.
- Jia, Z. & Han, X. (2010). Research and implementation of gis based on web. In *Computer application and system modeling (iccasm), 2010 international conference on* (Bd. 9, S. pages). doi:10.1109/ICCASM.2010.5622978
- Jiangfeng, S., Qian, C., Sen, P. & Xuezhi, F. (2009). Monitoring land use of construction based on webgis with ria technology. In *Information science and engineering (icise), 2009 1st international conference on* (S. 2104–2108). doi:10.1109/ICISE.2009.758
- JS, H. (2013). Interactive JavaScript charts for your web projects. Zugriff 21. Juni 2013, unter <http://www.highcharts.com/>
- Kamm. (2010). Sitepoint- thread: distance between long/lat point and line segment. Zugriff 10. Mai 2013, unter <http://www.sitepoint.com/forums/showthread.php?661462-Distance-between-long-lat-point-and-line-segment>
- Klopfers, M. (2005). *Interoperability and open architectures: an analysis of existing standardisation processes and procedures*. Open GIS Consortium Inc.
- Kompe, M. (2013). Entfernungsberechnung. Zugriff 15. Mai 2013, unter <http://www.kompf.de/gps/distcalc.html>
- Kowoma. (2008). Fehlerquellen bei GPS. Zugriff 28. Mai 2013, unter <http://www.kowoma.de/gps/Fehlerquellen.htm>
- Lange, N. (2007). *Geoinformatik in Theorie und Praxis*. Springer London, Limited. Zugriff unter <http://books.google.at/books?id=DijQflzkGEsC>
- Leaflet. (2013). An Open-Source JavaScript Library for Mobile-Friendly Interactive Maps. Zugriff 21. Juni 2013, unter <http://leafletjs.com/>
- LimeSurvey. (2013). The open source survey application. Zugriff 13. August 2013, unter <http://www.limesurvey.org/index.php/de/>
- Lobo, S. (2013). S.P.O.N. - Die Mensch-Maschine: Google macht die Welt zum Spielfeld. Zugriff 31. Mai 2013, unter <http://www.spiegel.de/netzwelt/web/google-ingress-die-ganze-welt-als-spiel-a-902267.html>
- Lu, X. (2006). Develop web gis based intelligent transportation application systems with web service technology. In *Its telecommunications proceedings, 2006 6th international conference on* (S. 159–162). doi:10.1109/ITST.2006.288823

Literatur

- Mabrouk, M., Bychowski, T., Niedzwiadek, H., Bishr, Y., Gaillet, J., Crisp, N., ... Margoulis, S. (2005). Opendata location services (opends): core services. *OGC Implementation Specification*, 5, 016.
- Miles, E. (2013). View Module. Zugriff 25. Juni 2013, unter <https://drupal.org/project/views>
- Minar, S. (2010). *Geoinformationssysteme (GIS) in der Wohnungs- und Immobilienwirtschaft*. Hammer, Patrick, Tanja Hammer, Matthias Knoop, Julius Mittenzwei, Georg Steinbach u. Michael Teltscher. GRIN Verlag GbR. Zugriff unter <http://books.google.at/books?id=TxswJRYIJIIC>
- Mitchell, T. (2008). *Web mapping mit open source-gis-tools* (first edition). O'Reilly.
- Nathansen, M. (2013). Grundprinzip der Satelliten-Navigation. Zugriff 10. April 2013, unter <http://gps.de/technik/>
- Neis, P. & Zipf, A. (2008). Zur kopplung von opensource, opensls und openstreetmaps in openrouteservice. org. In *Agit 2008, symposium für angewandte geoinformatik. salzburg. austria*.
- Netnode. (2012). Was ist Drupal? Zugriff 20. Mai 2013, unter <http://www.netnode.ch/was-ist-drupal>
- Netzwelt.de. (2013). Gadget-Globus: Schuh mit integriertem GPS-Empfänger. Zugriff 30. Mai 2013, unter <http://www.netzwelt.de/news/90581-gadget-globus-schuh-integriertem-gps-empfaenger.html>
- NianticLabs@Google. (2013). Ingress- the world around you is not what it seems. Zugriff 3. Juni 2013, unter <http://www.ingress.com/>
- OGC. (2013). Ogc history (abbreviated). Zugriff 8. April 2013, unter <http://www.opengeospatial.org/ogc/history>
- OpenLayers. (2012). Openlayers- javascript mapping library. Zugriff 15. Mai 2013, unter <http://dev.openlayers.org/docs/files/OpenLayers-js.html>
- OpenLayers. (2013). OpenLayers: Free Maps for the Web. Zugriff 21. Juni 2013, unter <http://openlayers.org/>
- OpenStreetMap. (2013a). OpenStreetMap- Die freie Wiki-Weltkarte. Zugriff 4. März 2013, unter <http://openstreetmap.org>
- OpenStreetMap. (2013b). OpenStreetMap- Die freie Wiki-Weltkarte. Zugriff 6. März 2013, unter <http://www.openstreetmap.org/edit>
- OpenStreetMap. (2013c). OSM-Lizenz. Zugriff 10. April 2013, unter <http://www.openstreetmap.org/copyright>
- Palazzolo, A. (2013). OpenLayers Module. Zugriff 26. Juni 2013, unter <https://drupal.org/project/openlayers>
- Peng, Z.-R. & Zhang, C. (2004). The roles of geography markup language (gml), scalable vector graphics (svg), and web feature service (wfs) specifications in the development of internet geographic information systems (gis). *Journal of Geographical Systems*, 6, 95–116. doi:10.1007/s10109-004-0129-0

Literatur

- Rostock, U. (2012). Geoinformatik-Lexikon. Zugriff 10. April 2013, unter <http://www.geoinformatik.uni-rostock.de/einzel.asp?ID=1841203769>
- Sandriesser, J. (2013). *Design and Implementation of an Gamification based Information System* (Magisterarb., TU Graz).
- Schütze, E. (2008). Uml class diagram of openlayers 2.7rc2 (pdf format). Zugriff 25. Juni 2013, unter http://trac.osgeo.org/openlayers/attachment/wiki/UML/ClassDiagram_OL2.7RC2-20080916.pdf
- Shen, L., Duan, W., Ren, Y. & Yang, C. (2010). Management service on wms/wfs services. In *Geoinformatics, 2010 18th international conference on* (S. 1–5). doi:10.1109/GEOINFORMATICS.2010.5567481
- Shreves, R. & Dunwoodie, B. (2011). *Drupal 7 bible*. Bible. Wiley.
- Stahl, F. & Schettler, O. (2012). *Praxiswissen drupal 7*. O'Reillys basics. O'Reilly Vlg. GmbH & Company.
- Strava. (2013). Strava — running and cycling gps tracker. Zugriff 26. April 2013, unter <http://www.strava.com/>
- SummitLynx. (2013). SummitLynx - Gipfelbuch. Zugriff 31. Mai 2013, unter <http://www.summitlynx.com/>
- Thiele, M. (2008). *Anwendungen Der Satellitennavigation in Der Logistik*. Hammer, Patrick, Tanja Hammer, Matthias Knoop, Julius Mittenzwei, Georg Steinbach u. Michael Teltscher. GRIN Verlag GbR. Zugriff unter <http://books.google.at/books?id=COKzECtLD64C>
- Van Diggelen, F. (2009). *A-gps: assisted gps, gnss, and sbas*. GNSS technology and applications series. Artech House, Incorporated. Zugriff unter <http://books.google.at/books?id=stTSHdFhrFUC>
- Weiser, A., Neis, P. & Zipf, A. (2006). Orchestrierung von OGC Web Diensten im Katastrophenmanagement. *GIS-Zeitschrift für Geoinformatik*, 9, 35–41.
- whitehouse.gov. (2012). How are you delivering on change. Zugriff 15. Mai 2013, unter <http://www.whitehouse.gov/change>
- Wien.at. (2013a). Open Government Data- Offene Daten für Wien. Zugriff 27. Mai 2013, unter <http://data.wien.gv.at/katalog/spielplaetze.html>
- Wien.at. (2013b). Schnittstellen Geowebsservices. Zugriff 6. März 2013, unter <http://data.wien.gv.at/formate/geowebsservices.html#wfs>
- Win, K. K. & Hla, K. H. S. (2005). Geospatial semantic query by integrating geospatial reasoning on the geospatial semantic web. In *Information and telecommunication technologies, 2005. apsitt 2005 proceedings. 6th asia-pacific symposium on* (S. 272–275). IEEE.
- CEN/TC287. (2013). Zugriff 8. April 2013, unter <http://www.centc287.eu/>
- Geography Markup Language. (2013). Zugriff 6. März 2013, unter <http://www.opengeospatial.org/standards/gml>
- Google Map Maker. (2013). Zugriff 10. April 2013, unter <http://www.google.com/mapmaker?hl=de>

Literatur

- ISO/TC211. (2013). Zugriff 8. April 2013, unter <http://www.isotc211.org/>
- Microformats. (2013). Zugriff 11. März 2013, unter <http://microformats.org/>
- OGC Members. (2013). Zugriff 15. April 2013, unter <http://www.opengeospatial.org/ogc/members>
- Open Geospatial Consortium. (2013). Zugriff 6. März 2013, unter <http://www.opengeospatial.org>
- OpenRouteService.org. (2013). Zugriff 12. April 2013, unter <http://openrouteservice.org/>
- OpenStreetMap. (2013). Zugriff 10. April 2013, unter <http://www.OpenStreetMap.org>
- OWL. (2013). Zugriff 11. März 2013, unter <http://www.w3.org/TR/owl2-overview/>
- RDF. (2013). Zugriff 11. März 2013, unter <http://www.w3.org/RDF/>
- Routing/online routers. (2013). Zugriff 12. April 2013, unter http://wiki.openstreetmap.org/wiki/Routing/OnlineRouters#Route_service_comparison_matrix
- Spielplätze - Standorte. (2013). Zugriff 11. März 2013, unter <http://data.wien.gv.at/katalog/spielplaetze.html>
- The Internet Engineering Task Force (IETF). (2013). Zugriff 11. März 2013, unter <http://www.ietf.org/>
- The OGC Process: How the Pieces Fit Together. (2013). Zugriff 6. März 2013, unter <http://www.opengeospatial.org/ogc/process>
- ViennaGIS. (2013). Zugriff 6. März 2013, unter <http://www.wien.gv.at/viennagis/>
- W3C. (2013). Zugriff 11. März 2013, unter <http://www.w3.org/>
- Web Feature Service. (2013). Zugriff 6. März 2013, unter <http://www.opengeospatial.org/standards/wfs>
- Web Map Service. (2013). Zugriff 6. März 2013, unter <http://www.opengeospatial.org/standards/wms>
- Wikimapia. (2013). Zugriff 10. April 2013, unter <http://wikimapia.org/>
- Yoshida, D., Song, X. & Raghavan, V. (2010). Development of track log and point of interest management system using free and open source software. *Applied Geomatics*, 2, 123–135. doi:10.1007/s12518-010-0028-7

Abbildungsverzeichnis

| | | |
|------|--|----|
| 2.1 | Schematische Darstellung der Funktionsweise eines WMS | 7 |
| 2.2 | Unterschied zwischen WMS und WFS | 9 |
| 2.3 | Unterschied zwischen Raster und Vektordarstellung | 12 |
| 2.4 | Aufbau eines hybriden Service | 13 |
| 2.5 | WebGIS Architektur | 14 |
| 2.6 | Ergebnis bei der Suche nach <i>Perth</i> | 15 |
| | | |
| 3.1 | Positionsbestimmung durch Satelliten | 18 |
| 3.2 | Arbeitsweise von A-GPS | 19 |
| 3.3 | Angabe einer Bar auf einer OSM Karte | 23 |
| 3.4 | OSM Statistik | 24 |
| 3.5 | Vergleich von OSM zu Google Maps | 25 |
| 3.6 | Rückgabe XML einer XAPI Anfrage | 28 |
| | | |
| 4.1 | Präsentation der Strecke auf <i>Runtastic</i> | 33 |
| 4.2 | Präsentation der Strecke auf <i>Endomondo</i> | 35 |
| 4.3 | Darstellung des <i>Ingress</i> Spielablaufs | 37 |
| | | |
| 5.1 | Darstellung des Systems, welches aus dem Webportal und der Datenspeicherung besteht. | 45 |
| 5.2 | Ablaufdiagramm des Erstellens einer Aktivität. | 46 |
| 5.3 | Zusammenspiel von Aktivitäten und Geocaching-Objekten im System. | 47 |
| | | |
| 6.1 | OpenLayers mit Bedienelementen | 50 |
| 6.2 | Verwendung von OpenLayers auf der OpenStreetMap Karte | 51 |
| 6.3 | Darstellung von vier Diagrammen in Highcharts | 52 |
| | | |
| 7.1 | Eine Drupal Request | 55 |
| 7.2 | OpenLayers mit Bedienelementen | 58 |
| 7.3 | Das Websystem in Drei-Schichten-Architektur | 59 |
| 7.4 | Ansicht einer Node in Drupal | 61 |
| 7.5 | Schematischer Ablauf des Erstellungsprozesses einer Strecke. | 62 |
| 7.6 | Das <i>OpenLayers</i> -Klassendiagramm mit den für die Implementierung wichtigsten Klassen | 63 |
| 7.7 | Streckenabschnitt mit vier Wegpunkten. | 66 |
| 7.8 | Ablauf der Routenerstellung | 66 |
| 7.9 | Schritte des Douglas Peucker Algorithmus | 67 |
| 7.10 | Veränderung des Streckenprofils | 69 |
| 7.11 | Karte mit Strecke in Verbindungen mit dem Höhendiagramm | 70 |
| 7.12 | Darstellung der Distanzberechnung mit sphärischen Koordinaten. | 71 |
| 7.13 | Entscheidungsfindung für Geocaching-Objekte. | 73 |

Abbildungsverzeichnis

| | | |
|------|--|----|
| 7.14 | Berechnung des Abstands eines Punkt-Objekts zu einer Strecke. | 75 |
| 7.15 | <i>OpenLayers</i> -Modul mit Geodaten. | 76 |
| 7.16 | Pop-ups und Streckenverlauf | 77 |
| 7.17 | Externer LayerSwitcher und Navigationshilfe | 78 |
| 8.1 | Übersichtskarte mit Geocaching-Objekten. | 80 |
| 8.2 | Formular für das Anlegen einer Aktivität. | 81 |
| 8.3 | Fertige Strecke erstellt mit dem Routenplaner. | 82 |
| 8.4 | Präsentation einer GPX-Datei. | 83 |
| 8.5 | Ändern des Streckenverlaufs durch Verschieben von Wegpunkten. | 84 |
| 8.6 | Ansicht einer Aktivität mit Geocaching-Objekt. | 85 |
| 8.7 | Ansicht der Rennstrecke und des Rankings der einzelnen User. | 86 |
| 8.8 | Tortendiagramm mit der Häufigkeit des Auftretens der verschiedenen Sportarten. | 88 |
| 8.9 | Statistik der Häufigkeit der Antworten auf die Frage, welche Funktionen bei Software zur Kontrolle von sportlichen Leistungen verwendet wird. | 89 |
| 8.10 | Statistik der Häufigkeit der Antworten auf die Frage, welche Funktionen bei Software zur Kontrolle von sportlichen Leistungen sich motivierend auf Leistungen auswirken. | 90 |
| 8.11 | Zusammenfassung der drei Fragen bezüglich Funktion und Performance des Routenplaners. | 91 |

Tabellenverzeichnis

- 7.1 Aufstellung des Arrays, welches die Wegpunkte zur Darstellung der Strecke hält 65
- 7.2 Tabellarische Darstellung der Wegpunkte nach dem Parsen aus einer GPX-Datei 68

Listings

| | | |
|-----|--|----|
| 2.1 | Auszug einer von Vienna GIS bereit gestellten GML-Datei (Wien.at, 2013a) . . . | 11 |
| 2.2 | Drei verschiedene geografische Objekte, welche im Well-Known-Text Format beschrieben werden. | 11 |
| 2.3 | Aufbau einer Worlddatei, welche die Position und Lage eines Rasterbildes bestimmt. | 12 |
| 3.1 | Aufbau der API für eine Nominatim Anfrage | 25 |
| 3.2 | Die XML Rückgabe einer Nominatim Anfrage zeigt die Vielzahl an Informationen, welche für ein Koordinatenpaar vorhanden sind. | 26 |
| 3.3 | Aufbau der API für eine Reverse-Nominatim Anfrage | 26 |
| 3.5 | XML Rückgabe einer Nominatim Anfrage | 27 |
| 3.4 | XAPI Anfrage für Point-of-Interest in einer definierten Box. | 27 |
| 7.1 | Der Konstruktor für das Erstellen eines Map-Objekts | 56 |
| 7.2 | Dieser Codeauszug zeigt den Konstruktor für ein WMS-Layer Objekt. | 57 |
| 7.3 | Konstruktoren der wichtigsten OpenLayers Control-Objekte. | 58 |
| 7.4 | Erstellen eines Vector-Layers mit Angabe von Style-Informationen. | 63 |
| 7.5 | Dieser Ausschnitt aus einer GPX-Datei zeigt zwei Track-Point Elemente innerhalb eines Treck-Segments. | 67 |
| 7.6 | Berechnung der Distanz zwischen Koordinaten | 70 |

Anhang B

Inhalt der beiliegenden CD-ROM

Die beiliegende CD-ROM enthält alle für diese Masterarbeit wichtigen Daten aufgeteilt in die Ordner *Source* und *Dokumente*.

Source

- Datenbank
- Source-Code

Dokumente

- Literatur: Papers und Dokumente, welche als Quellen des theoretischen Teils dieser Arbeit verwendet wurden.
- Umfrage: Die Aufstellung des Fragebogens, sowie alle Antworten einzeln und zusammengefasst.
- Masterarbeit: Diese Masterarbeit in PDF Format.