

**Philosophische** Fakultät III

Sprach- , Literatur- und Kulturwissenschaften

Institut für Information und Medien, Sprache und Kultur (I:IMSK)  
Lehrstuhl für Medieninformatik

Einführung in die Anwendungsprogrammierung

Modul: MEI-M03.3

Sommersemester 20XX

Garbage Collectors  
Wann kommt die Müllabfuhr?

[Autor der Arbeit]

[Matrikelnummer]

[Studienfächer]

[Semesterzahl und Studiengang (z.B. 3. Semester M.A.)]

[Straße mit Hausnummer]

[Postleitzahl mit Wohnort]

Tel.: [Telefonnummer (z.B. 0941/9999)]

Email: [Emailadresse (z.B.: max.mustermann@stud.uni-regensburg.de)]

Abgegeben am [Abgabetermin der Arbeit]

Inhalt

1 Anmerkungen zu dieser Vorlage 4

2 Kurzbeschreibung 5

3 Technische Vorrausetzungen 6

4 Problemstellung und Lösungsvorschlag 7

5 Design & Implementierung 7

5.1 User Interface 7

5.2 Implementierung 7

6 Testing 7

7 Finaler Zustand und Ausblick 7

*8* Projektmanagement 7

Abbildungen

Abbildung 1: Startbildschirm der Anwendung 7

# Anmerkungen zu dieser Vorlage

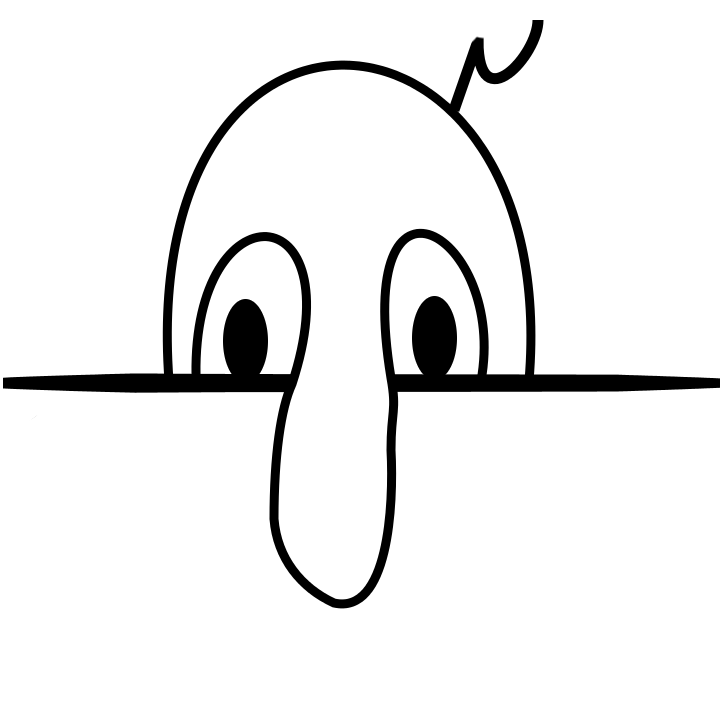
Diese Vorlage dient der Dokumentation der Abschlussprojekte des Android-Kurses. Sie basiert auf der allgemeinen Vorlage für Seminararbeiten und Abschlussarbeiten der Lehrstühle für Medieninformatik und Informationswissenschaft. Bitte benutzen Sie die vorgegebenen Kapitel als Muster für die Anfertigung Ihrer Projektbeschreibung; jedes Kapitel beinhaltet dazu eine kurze Zusammenfassung der erwarteten Inhalte. Löschen Sie vor der Abgabe der Dokumentation diese und andere Anmerkungen aus dem Dokument (ebenso dieses Kapitel). An einigen Stellen enthält die Vorlage Beispieltexte für die fiktive App *Garbage Collectors.* **Diese Texte dienen als Formulierungshilfe und Anregung für Ihre eigene Projektbeschreibung und stellen keine komplette Dokumentation dar.** Bitte reichern Sie Ihre Ausführungen durch sinnvolle Graphiken und Screenshots an.

Es gibt keine vorgegebene Seitenanzahl! Formulieren Sie die nötigen Informationen umfassenden, prägnant und verständlich. Die verschiedenen Kapitel können (und sollten) Sie sinnvoll durch Unterkapitel gliedern.

Neben dieser Dokumentation sind noch weitere Projektbestandteile abzugeben. Die genauen Anforderungen entnehmen Sie bitte den entsprechenden Folien zur Vorlesung.

# Kurzbeschreibung

Beschreiben Sie in diesem Kapitel kurz die wesentlichen Features Ihrer App und begründen Sie, warum ein Benutzer Ihre Anwendung installieren und nutzen sollte. Versuchen Sie dabei einen kurzen Text zu entwerfen, den Sie auch im Google Playstore verwenden könnten. Die Kurzbeschreibung darf in keinem Fall länger als eine Seite sein.

Der Name Kilroy stammt aus den 1940er und 1950er Jahren und wurde dort als Graffito von US-Soldaten benutzt. Die Formulierung „Kilroy was here“ galt unter ihnen als running Gag. Der Slogan wurde oft von einem Bild, das ein Gesicht mit einer langer Nase und zwei großen Augen zeigte, begleitet. Unter den Soldaten brach ein Wettbewerb aus, das Graffito an den unmöglichsten und entlegensten Orten anzubringen. Wegen dieser Parallele zwischen dem Graffito und der App wählten wir den Namen Kilroy.

Die App Kilroy ermöglicht es digitale Gästebücher zu erstellen und in der realen Welt zu platzieren. Gästebücher können nur geöffnet werden wenn sich der jeweilige Nutzer auch tatsächlich am Standort des Buches befindet. Dadurch wird dem Gästebuch eine örtliche Präsenz verliehen. Ermöglicht wird das durch NFC-Tags, kleine beschreibbare Nahfeldkommunikations-Chips die als Sticker, Anhänger und in vielen weiteren Formen erhältlich sind. Sie repräsentieren ein Gästebuch in der realen Welt. Ein NFC-Tag kann nun also an einem bestimmten Ort platziert und mithilfe der Kilroy App gescannt werden. Dadurch wird ein neues Gästebuch an der aktuellen Position des Nutzers erstellt. Andere Nutzer haben nun die Möglichkeit diesen Tag mithilfe einer Integrierten Karte zu finden und ebenfalls zu scannen. Dadurch wird sich das zuvor erstellte Gästebuch öffnen. Der Nutzer kann nun einen oder mehrere Einträge für zukünftige Besucher, oder auch den Besitzer des Gästebuchs zu hinterlassen.

# Technische Vorrausetzungen

Nennen und beschreiben Sie in diesem Kapitel die technischen Voraussetzungen die für den Betrieb bzw. die Installation Ihrer App notwendig sind. Sie sollten in diesem Kapitel darlegen, welche externen Anwendungen (Server) für die Nutzung Ihrer Applikation nötig sind und wo bzw. wie diese heruntergeladen, eingerichtet und gestartet werden können. Mit diesen Informationen muss es möglich sein, Ihre Anwendung in vollem Umfang zu testen und etwaige Abhängigkeiten aufzulösen bzw. notwendige Infrastruktur-Anwendung in Betrieb zu nehmen. Zusätzlich erwähnen Sie hier externe Bibliotheken und Frameworks, die Sie für die Entwicklung Ihrer Anwendung genutzt haben. Geben Sie dabei auch an, für welche Funktionen oder Komponenten die jeweiligen Bibliotheken eingesetzt werden.

Kilroy ist eine Mobile-Anwendung mit eigenem Webservice der die Gästebücher Online in einer MySql Datanbank [https://www.mysql.com] speichert und somit das Backend der App darstellt. Der Webservice ist mithilfe des Spark Frameworks [http://sparkjava.com] in Java implementiert. Für die Kommunikation mit dem MySql Server wird die Bibliothek sql2o [http://www.sql2o.org] verwendet. Außerdem wurde als Webserver ein Cloudserver von OpenShift [https://www.openshift.com/] verwendet. Die Daten der App werden mit dem Backend mittels HTTP im Json Format ausgetauscht und anschließend mithilfe von Gson [https://sites.google.com/site/gson/Home] geparst. Die Hostadresse des Webservice lautet: „http://kilroybackend-kilroybackend.rhcloud.com/api/“.

Um das Backend lokal zu starten ist eine lokale Installation eines MySql Servers, der unter der Hostadresse „localhost“ sowie dem Port „3306“ erreichbar ist, erforderlich. Die Datenbank kann mithilfe des MySql Dump „kilroy\_db\_dump.sql“, der sich Verzeichnis „KilroyBackend“ befindet, eingespielt werden.

Voraussetzungen für die Nutzung der App ist ein Android-Gerät welche mindestens die Version 4.2.2 des Android-Systems Nutzen kann. Außerdem muss eine aktive Internetverbindung, NFC und GPS verfügbar sein. Unterstützt werden Smartphones, Tablets hingegen nicht. Zusätzlich werden NFC-Tags zur Erstellung von eigenen Gästebüchern benötigt.

Um den Lese- und Schreibzugriff auf NFC-Tags zu erleichtern, wurde die „ndef-tools-for-android“ [https://github.com/skjolber/ndef-tools-for-android] genutzt. Diese stellt eine high-level API für den Umgang mit NFC-Technologie zur Verfügung.

*(Garbage Collectors* ist eine reine Mobil-Anwendung, die ohne eigene Server-Infrastruktur funktioniert. Die nötigen Daten werden direkt von der Website des Regensburger Entsorgungsunternehmens Meindl [http://www.entsorgungsdaten.de/] bezogen. Für den Betrieb der Anwendung reicht eine (manuelle) Installation der Anwendung auf einem Android-Gerät aus. Unterstützt werden Smartphones und Tablets, die mindesten die Version 4.0 des Android-Systems nutzen und über eine aktive Internetverbindung verfügen

Für die Datenbeschaffung werden die entsprechenden Unterseiten der Meindl-Website bezogen und ausgewertet. Dazu wird der HTML-Code der jeweiligen Seiten über eine aktive Netzwerkverbindung heruntergeladen und anschließend ge-*parst*. Für diesen Vorgang wird die JAVA-Bibliothek jsoup [jsoup.org] in der Version 1.7.3 eingesetzt. Jsoup erlaubt das einfache und komfortable *Parsen* von HTML-Strukturen innerhalb von JAVA-Anwendungen. Jsoup wird hier für das schnelle Heraussuchen von HTML-Elementen aus einer gegeben Website genutzt. Dabei wird auf komplexe CSS-Selektoren zurückgegriffen, die eine gezielte Auswahl einzelner Knoten innerhalb komplexer HTML-Strukturen erlauben. Dadurch können die nötigen Informationen (Entsorgungstermine) schnell und einfach aus der restlichen Website herausgefiltert werden. [...])

# Problemstellung und Lösungsvorschlag

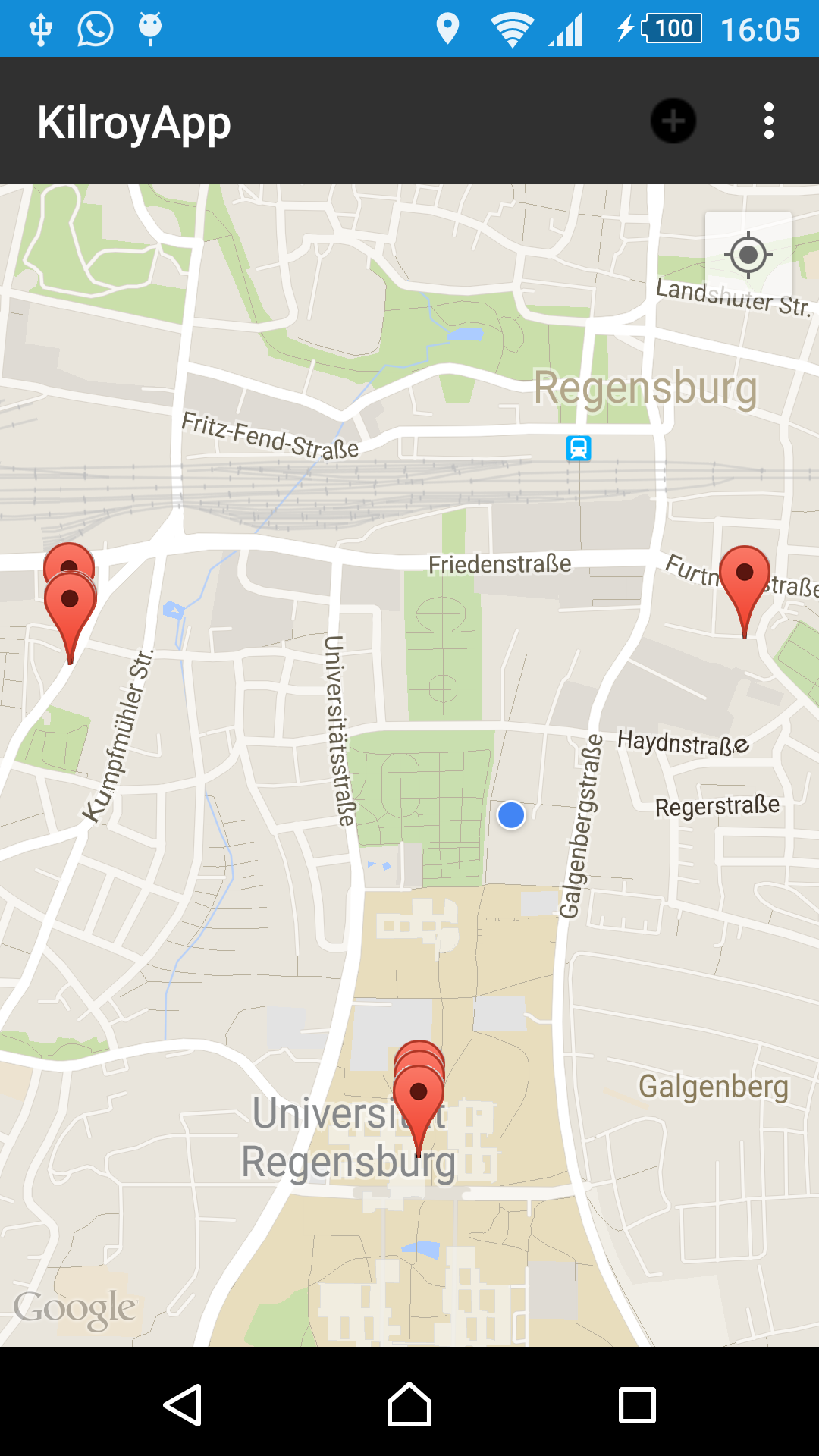
Beschreiben Sie in diesem Kapitel den konkreten Anwendungsfall, der Ihrer App zu Grunde liegt. Welches Problem löst Ihre Anwendung und in welchem Szenario ist sie einsetzbar? Dieses Kapitel sollte die Motivation für Ihr Projekt darlegen und beschreiben, welchen grundlegenden Lösungsansatz Sie für die gegebene Problemstellung vorschlagen. Achtung: Sollten Sie ein Spiel umsetzten, kann dieses Kapitel möglicherweise schwierig zu formulieren sein. Beschreiben und motivieren Sie in diesem Fall die grundlegende Spielidee.

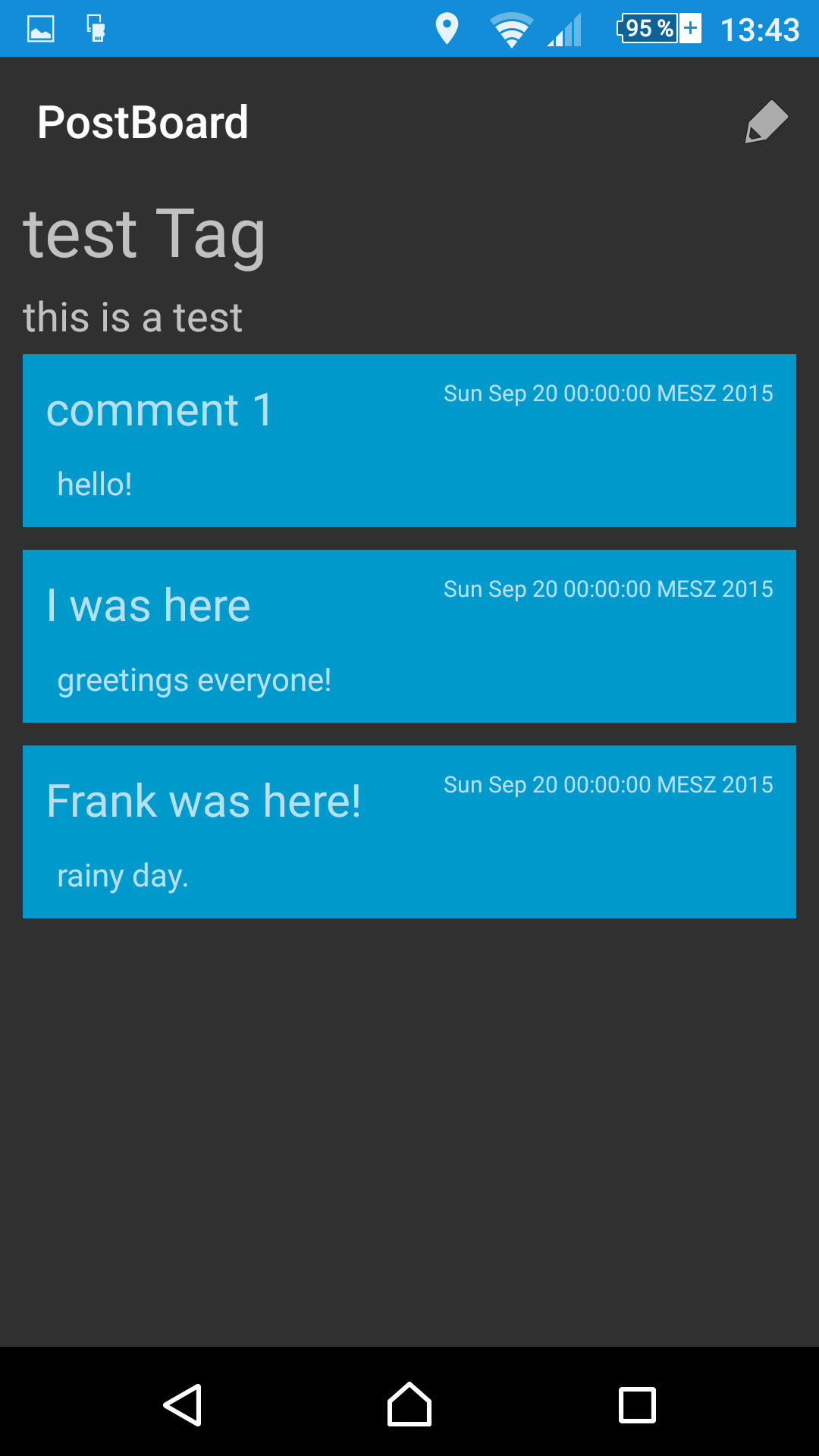
Besucht man Touristen Attraktionen oder andere besondere Orte in einer Stadt oder In der Natur, stößt man des Öfteren auf Nachrichten der vorhergehenden Besucher. Wie etwa „wir waren hier 2013“ oder auch „Jana und Mike 2014“. Diese Nachrichten und die damit verbundene soziale Komponente soll auf eine digitale Version übertragen werden. Ebenso sollte der Erkundungsdrang sowie die Neugier darauf, welche Nachrichten an einem bestimmten Ort hinterlassen wurden geweckt werden. Inspiriert wurde diese Idee unter anderem von Anwendungen wie Geocaching und das von Niantic entwickelte virtuell-reality Spiel *Ingress.* Um die Verfügbarkeit solcher Nachrichten auf einen bestimmten Ort zu beschränken, war die NFC-Technologie besonders geeignet. Diese erlaubt es, Daten auf günstig zu erwerbenden Chips, bei einer maximalen Entfernung von 5cm, zu schreiben und auch wieder zu lesen.

Um diese Daten sowie die Position der NFC-Tags für andere Nutzer sichtbar zu machen werden diese auf einem Server gespeichert und bei Bedarf auf das jeweilige Gerät geladen. Ein weiterer Faktor besteht in der Anonymität des Nutzers. So soll er wie auch beim hinterlassen von Nachrichten in der realen Welt, innerhalb der App anonym bleiben. Gästebücher oder Kommentare erhalten deshalb keinen Bezug auf deren Verfasser.

# Design & Implementierung

In diesem Kapitel sollen Sie die grundsätzlichen Konzepte der Designphase und Ihr Vorgehen während der Implementierung der Anwendung darlegen. Beschreiben Sie dazu die Architektur, die Sie sich für die Anwendung überlegt haben und dokumentieren Sie, wie Sie bezüglich der Implementierung der verschiedenen Funktionen vorgegangen sind. Dem Leser sollte deutlich werden, welche Komponenten und Module Sie für die Anwendung geplant haben und wie Sie bei deren Umsetzung vorgegangen sind. Beschreiben Sie dazu, aus welchen Activities bzw. Ansichten Ihre Anwendung besteht, welche Aufgabe diese einzelnen Komponenten haben und wie die Kommunikation bzw. Interaktion zwischen diesen Bestandteilen aussieht. Erwähnen Sie dabei auch mögliche – externe – Datenquellen und deren Anbindung. Beschreiben Sie die wichtigsten Klassen die Sie für die Implementierung der verschiedenen Funktionalitäten eingesetzt bzw. geschrieben haben und erwähnen Sie mögliche Probleme und Lösungsansätze. Beschreibend und begründen Sie kurz, in welchen Reihenfolge die verschiedenen Komponenten der Anwendung umgesetzt wurden.

Die Anwendung besteht aus fünf Activities. Einem Home Screen der als Einstiegspunkt der App dient. Von hier aus kann der Nutzer zur Übersichtskarte, Hilfefunktion oder der Activity zum Erstellen neuer Gästebücher gelangen. Auf der Übersichtskarte werden alle bereits erstellten Gästebücher durch Marker angezeigt. Klickt man auf einen solchen Marker wird dessen Name angezeigt. Klickt man wiederum auf den Namen des Markers, startet eine Activity (MarkerDetailActivity) welche die Beschreibung zum Auffinden desselben beinhaltet. In der Hilfe Activity findet der Nutzer eine Beschreibung der App sowie Anweisungen zum Erstellen von Gästebüchern und Kommentaren.

Innerhalb der Gästebuch anlegen Activity können Name und Beschreibung des zu erstellenden Gästebuchs eingetragen werden und anschließen mit einem Klick auf den entsprechenden Button auf einen NFC-Tag geschrieben werden und die Daten werden an den Server gesendet. Dadurch wird das erstellte Gästebuch für alle andern Nutzer sichtbar und aufrufbar. Scannt ein Nutzer einen entsprechend beschriebenen NFC-Tag gelangt er in die PostBoard Activity. Diese bietet ihm die Möglichkeit Kommentare in dem geöffnetem Gästebuch zu hinterlassen. Außerdem werden dort bereits erstellte Kommentare von ihm und andern Nutzern angezeigt. Ein Klick auf das Stiftsymbol im oberen rechten Bereich öffnet wiederum eine Aktivity zum Erstellen eines Kommentars. Hier können Autor und Inhalt des Kommentars eingetragen und gespeichert werden. Da wir Wert auf Anonymität legen, steht dem Nutzer die Wahl des Pseudonyms frei.

Neben den Sichtbaren Komponenten verfügt die App über einen AppControler. Dieser regelt die Anfragen (Get- und Post-Request) an den Server. Über ein Listener-Pattern werden die *CommentActivity, KilroyNfcTagWriterActivity, MainActivity* und *PostboardActivity*  an den Controller gebunden. Über die Listener-Schnittstelle *onResponse* erhalten die Activitys die angeforderten Daten als Json-Objekte. Diese werden mithilfe von Gson geparst und in Java-Objekte überführt.

Die Anwendung besteht aus drei Activities. Eine Übersichtsdarstellung (OverviewActivity) dient als Einstiegspunkt der App und stellt den nächsten Entsorgungstermin für die als Favorit markierte Straße dar. Durch einen Wisch über den Bildschirm wechselt der Nutzer zu eine weitere Activity (StreetL MarkerDetailActivity istActivity), die die Suche nach verschiedenen Straßen erlaubt. Die angezeigt List enthält – alphabetisch sortiert – alle verfügbaren Straßen. Über ein einblendbares Suchinterface kann gezielt nach einem Straßennamen gesucht werden. Die angezeigt Liste wird dabei – automatisch – in Abhängigkeit der jeweiligen Nutzereingabe gefiltert. Durch den Klick auf einen Listeneintrag gelangt der Nutzer zu einer Detailansicht der ausgewählten Straße (StreetDetailActivity), die die Entsorgungstermine für die jeweilige Straße enthält. Diese Termine sind aufsteigend sortiert und in einer Liste angeordnet. Jedes Item der List enthält den entsprechenden Termin sowie die Art des Abfalls (Restmüll, Papier, ...). Über ein Icon der Actionbar kann die aktuell angezeigt Straße als Favorit gesetzt werden. [...]

Neben den sichtbaren Komponenten verfügt die Anwendung über eine Reihe von Modulen, die zur Beschaffung der nötigen Daten eingesetzt werden. Eine zentrale Kontroller-Einheit (AppController) kapselt dabei den Zugriff auf die Daten und sorgt für deren Aktualisierung. Andere Komponenten können über ein Listener-Pattern an diesen Kontroller angebunden werden. Der Kontroller baut die Verbindung mit den nötigen Internetseiten auf und verarbeitet die angeforderten Webseiten mit den Entsorgungsterminen. Die ge-*parsten* Inhalte werden in JAVA-Objekte überführt und über die Listener-Schnittstelle zur weiteren Verwendung innerhalb der Anwendung bereit gestellt. [...]

## User Interface

Beschreiben Sie an dieser – oder einer anderen sinnvollen – Stelle den Aufbau des User Interfaces. Aus welchen Komponenten (Views) bestehen die einzelnen Activities und wie kann der Benutzer mit Ihnen interagieren?

Abbildung : Startbildschirm der Anwendung

## Implementierung

Die wesentliche Aufgabe der App besteht darin den Nutzern die Positionen und Inhalte der Gästebücher bereitzustellen. Parallel zur Implementierung des Backends wurden die *PostboardActivity,*  zur Darstellung des Gästebuchs und dessen Kommentare, und die  *MainActivity,* welche eine Übersichtskarte mit den Positionen der Gästebücher anzeigt. Nach Fertigstellung des Backends wurden auf beiden Seiten die jeweiligen Adapter zum Austausch der Daten implementiert.

Die wesentliche Aufgabe der Anwendung besteht in der Bereitstellung der Entsorgungsdaten. Initial wurde daher die Verarbeitung bzw. die Beschaffung dieser Daten getestet und implementiert.

Die nötigen Daten konnten auf der Website der Firma Meindl identifiziert werden. Im Wesentlichen finden sich die nötigen Angaben auf zwei Unterseiten des Internetauftritts. Eine Liste[[1]](#footnote-1) enthält alle Straßennamen, die von dem Entsorgungsbetrieb angefahren werden. Der HTML-Code dieser Seite beinhaltet für jede Straße einen Link zu einer Detailansicht[[2]](#footnote-2), auf der die eigentlichen Entsorgungsdaten aufgeführt werden. In einem ersten Schritt wurde die Übersichtseite (Straßenliste) bzw. der HTML-Code dieser Seite geladen. Dazu wurde die entsprechende URL über einen HTTP-Request (über den HTTP-Client des Apache-Pakets) aufgerufen. Der zurückgegebene HTML-Code der Website konnte anschließend durch die Verwendung der jsoup-Bibliothek weiter verwendet werden. Eine vorherige Analyse der HTML-Struktur zeigte, das die Liste der Straßennamen bzw. deren einzelne Bestandteile über verschiedene CSS-Klassen identifiziert werden können. Ein entsprechend implementierter Parser konnte über die einzelnen Straßen iterieren und die wesentlichen Bestandteile herausfiltern. Der Straßenname, die Postleitzahl sowie eine eindeutige ID – vergeben durch die Website – wurden in eine eigene JAVA-Repräsentation (Street-Klasse) überführt. Der Link zu der detaillierten Auflistung der Entsorgungstermine ist für alle Straßen gleich und unterscheidet sich nur durch die gespeicherte ID. Als Ergebnis dieses ersten Schrittes konnte eine Liste alle Regensburg Straßen - in Form von JAVA-Objekten – erstellt werden, für die Entsorgungsdaten vorliegen. [...]

In einem weiteren Schritt wurde die Detailansicht der jeweiligen Straßen verarbeitet. Über die bekannte URL – mit ergänzter, individueller Straßen-ID – wurde der HTML-Code bezogen und durch Einsatz der jsoup-Bibliothek verarbeitet. [...]

Nach erfolgreichem Test der Datenanbindung wurde die erarbeitet Funktionalität in wiederverwendbarem, integrierbarem Code umgesetzt. Zentrale Komponenten dazu ist der DataController, der den Aufruf und die Verarbeitung der verschiedenen Webseiten steuert und die erstellen Datensätze (Straßen und jeweilige Entsorgungsdaten) über ein Interface bereitstellt. Andere Komponenten können den Kontroller instanziieren; über den Konstruktor wird dabei eine Referenz auf einen OnDataChangedListener (Interface) übergeben. Über öffentliche Methoden des Kontrollers kann die Aktualisierung der Straßen-Liste (update()) sowie die Beschaffung von Entsorgungsterminen für eine bestimmte Straße (fetchGarbageDaysFor Street(Street street)) ausgelöst werden. Der Listener wird vom Kontroller benachrichtigt, wenn eine Aktualisierung der Datensätze erfolgreich war. Über die entsprechenden Methoden des Interfaces werden Listen an verfügbaren Straßen (Street) bzw. Entsorgungstermine (GarbageDay) übermittelt. [...]

Nach der Fertigstellung der Komponenten zur Datenbeschaffung wurde die Benutzer-Schnittstelle der Anwendung implementiert. Dazu wurden die geplanten und entworfenen Activities umgesetzt. [...] Für die Darstellung der Straßenliste innerhalb der StreetListActivity wurden ein Adapter sowie ein XML-Layout entworfen, der eine Liste von Street-Objekten mit einem ListView verknüpft und die relevanten Inhalte darstellt. [...]

# Testing

Beschreiben Sie die wesentlichen Probleme, die durch den Usability-Test aufgedeckt wurden. Erläutern Sie kurz, welche Maßnahmen ergriffen wurden, um diese Problem zu beheben und geben Sie an, im welchem Umfang die identifizierten Schwachstellen ausgebessert wurden. Beschreiben Sie, wie Sie Ihre Anwendung angepasst haben, um die identifizierten Problem zu lösen. Dabei sollten Sie erwähnen welche Komponenten oder Klassen Ihren App betroffen sind und wie sich die Änderungen auf Ihr ursprüngliches (Interaktions-)Konzept ausgewirkt haben.

Während des Usability-Tests wurde unter anderem das Fehlen einer Hilfe-Activity als Problem erkannt. Ebenso hatten Nutzer Schwierigkeiten beim Erstellen eines neuen Gästebuchs. Dies lag unteranderem daran, dass sich der Button zum Erstellen eines Gästebuchs im Options-Menü befand. Auch der Vorgang zum Beschreiben eines NFC-Tags gestaltete sich für die Mehrheit der Nutzer als schwierig. Sollte ein Kommentar in einem Gästebuch erstellt werden, bemängelten viele Teilnehmer des Tests die Platzierung des Buttons der zum Speichern eines Kommentars gedrückt werden musste. Dieser befand sich im rechten Bereich der Actionbar.

Um diese Probleme zu lösen und die Usability der App zu verbessern, wurde ein Hilfescreen erstellt. Der Button zum Erstellen eines Gästebuchs wurde als Icon in die Actionbar eingefügt. Der Vorgang zum Beschreiben eines NFC-Tags wurde in mehrere Einzelschritte untergliedert. Dies soll den Vorgang zum Erstellen eines neuen Gästebuchs übersichtlicher gestalten.

Für einen weicheren Einstieg in die App wurde ein Mainscreen, der Buttons zur Navigation durch die App bietet. Da jedoch die Kartenansicht bei der App eine zentrale Rolle spielt, wurde diese weiterhin als Einstiegspunkt in die App beibehalten. Des Weiteren wurde empfohlen, die Detail-Activity, welche Namen und eine genaue Beschreibung zum Auffinden des Gästebuchs beinhaltet, zu entfernen. Da jedoch das Finden von Gästebüchern im Usability-Test nicht berücksichtigt wurde, blieb die Detail-Activity in der App. Denn ohne sie und die darin enthaltene Beschreibung, wird das Auffinden eines Gästebuchs unnötig erschwert. Die mehrere Zeilen lange Beschreibung im kleinen Infofenster des Markes zusammen mit dem Namen und das daraus resultierende, unübersichtliche Layout war ein weiterer Grund für das Beibehalten der Detail-Activity.

# Finaler Zustand und Ausblick

Beschreiben Sie kurz den finalen Zustand Ihrer Anwendung. Erläutern Sie den implementierten Funktionsumfang und begründen Sie fehlende Features, die in den Anforderungen erwähnt wurden, in der finalen Version jedoch nicht verfügbar sind. Überlegen Sie, inwieweit Ihre Anwendung durch zusätzliche Funktionen sinnvoll erweitert werden könnte.

Im finalen Zustand der App ist es möglich Gästebücher auf der Karte zu erstellen und dort Einträge zu hinterlassen. Ebenso ist es möglich bereits vorhandene Gästebücher zu finden. Auf eine Nutzerregestierung wurde bewusst verzichtet, um die Anonymität der Nutzer zu gewährleisten.

Für die Zukunft bietet sich das Feature an, Bilder vom Standort der Gästebücher hinzuzufügen. Diese sollen helfen, den Tag leichter zu finden. Auch eine Favoriten Liste von oft besuchten Gästebüchern wäre denkbar. Diese Features erweitern die App sinnvoll, jedoch werden sie nicht für die Kernfunktionalität der App benötigt.

# Projektmanagement

Beschreiben Sie kurz, wie Sie bezüglich der gruppeninternen Abstimmung vorgegangen sind. Erwähnen Sie Tools und Kommunikationswege, die Sie für die Gruppenarbeit eingesetzt haben. Sie können hier auch einen kurzen Überblick über die Arbeitsaufteilung geben und explizit erwähnen, welche Bestandteile von welchen Teammitgliedern umgesetzt wurden.

Vorschläge, Änderungen und Anregungen wurden stets bei gemeinsamen Treffen vor Ort diskutiert und festgehalten. Ging es nicht anders wurde über Skype kommuniziert. Zur Verwaltung des gemiensamen Codes wurde ein Github Repository erstellt. Hier wurden Änderungen und Erweiterungen auf einen Development-Branch gepusht und beim erreichen von Stabilen Versionen in den Master-Branch gemered.

1. http://www.entsorgungsdaten.de/kalender/ [↑](#footnote-ref-1)
2. Beispiel: http://www.entsorgungsdaten.de/module.php5?fid=15&gem=30&mod=entsorgungsdaten&ot=1047 [↑](#footnote-ref-2)