



Universität Regensburg

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 2015

KilroyApp

Florian Pils, Hannes Heuberger, Simon Geyer
Pils 1726554, Heuberger 1739610, Geyer 1755523
MI / MW, MI / INF, MI / INF
2. Semester BA

Simon Geyer
Bischof-Wittmann-Str. 10b
93051 Regensburg
Tel.: 0163 / 7618341
Email: ges62042@stud.uni-regensburg.de, study@simongeyer.com
Abgegeben am 28.09.15

Inhalt

1	Anmerkungen zu dieser Vorlage	3
2	Kurzbeschreibung	4
3	Technische Vorrausetzungen	5
3.1.1	Kilroy App	5
3.1.2	Kilroy Backend	5
3.1.3	Github	5
4	Problemstellung und Lösungsvorschlag	6
5	Design & Implementierung	7
5.1	Implementierung	10
5.1.1	Kilroy App	10
5.1.2	Kilroy Backend	11
6	Testing	12
7	Finaler Zustand und Ausblick	13
8	Projektmanagement	14

1 Anmerkungen zu dieser Vorlage

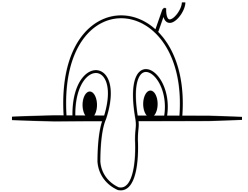
Diese Vorlage dient der Dokumentation der Abschlussprojekte des Android-Kurses. Sie basiert auf der allgemeinen Vorlage für Seminararbeiten und Abschlussarbeiten der Lehrstühle für Medieninformatik und Informationswissenschaft. Bitte benutzen Sie die vorgegebenen Kapitel als Muster für die Anfertigung 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 Beispieltex-te für die fiktive App *Garbage Collectors*. **Diese Texte dienen als Formulierungshilfe und Anregung für Ihre eigene Projektbeschreibung und stellen keine komplette Dokumentation dar.** Bitte bereichern 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.

2 Kurzbeschreibung

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.

3 Technische Voraussetzungen

Kilroy ist eine Mobile-Anwendung mit eigenem Webservice der die Gästebücher Online in einer MySQL Datenbank [<https://www.mysql.com>] speichert und somit das Backend der App darstellt.

3.1.1 Kilroy App

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.

3.1.2 Kilroy Backend

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>] geparkt. Die Hostadresse des Webservices 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.

3.1.3 Github

Der Source Code des Backends [<https://github.com/sabralod/kilroybackend.git>] und der App [<https://github.com/sabralod/kilroyapp.git>] ist außerdem auf Github verfügbar.

4 Problemstellung und Lösungsvorschlag

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.

5 Design & Implementierung

Die Anwendung besteht aus mehreren Activities. Die MainActivity bindet das GoogleMap Widget ein und stellt den Einstiegspunkt der App dar. Von hier erhält der Nutzer Zugriff auf alle wichtigen Funktionen der App. 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 die *MarkerDetailActivity*, welche die Beschreibung zum Auffinden desselben beinhaltet.



Abbildung 1: Start Dialog

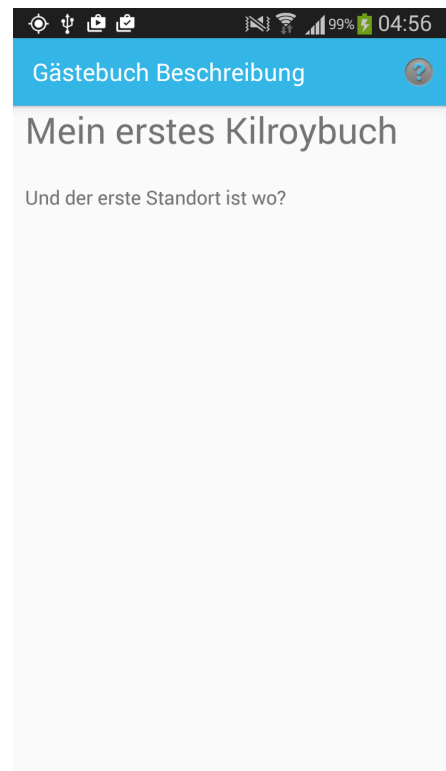


Abbildung 2: Gästebuch Infos

Klickt man auf das Fragezeichen Icon, gelangt man in die *HelpActivity*. Hier findet der Nutzer eine Beschreibung der App sowie Anweisungen zum Erstellen von Gästebüchern und Kommentaren. Innerhalb der *CreatePostActivity* wird ein Gästebuch erstellt. Name und Beschreibung des zu erstellenden Gästebuchs werden eingetragen

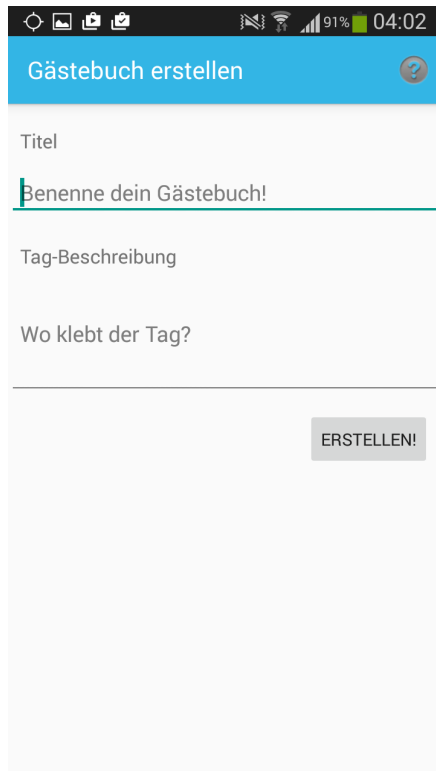


Abbildung 3: Gästebuch erstellen

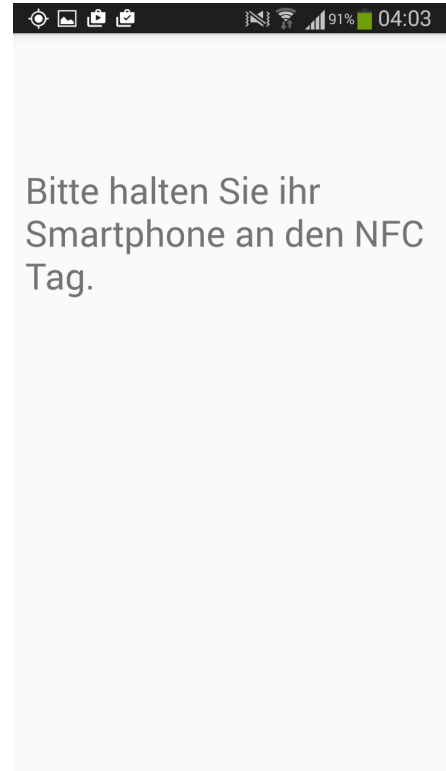


Abbildung 4: NFC Tag berühren

und anschließend mit einem Klick auf den ERSTELLEN! Button in der *WriterActivity* auf einen NFC-Tag geschrieben und 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 *PostBoardActivity*.

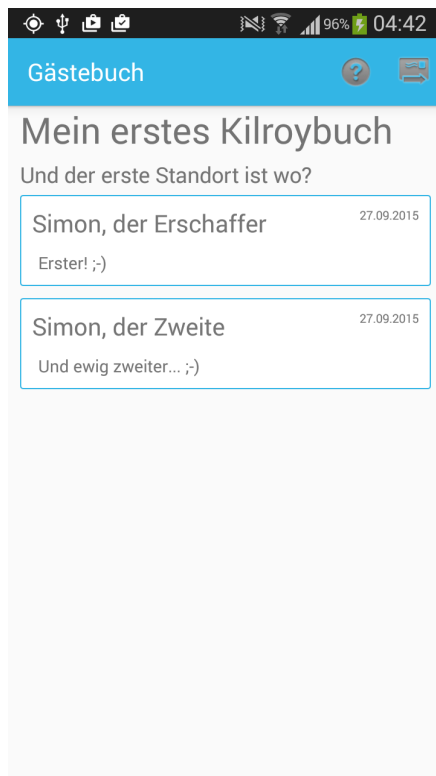


Abbildung 5: Gästebuch lesen



Abbildung 6: Kommentar schreiben

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 Briefsymbol im oberen rechten Bereich öffnet die *CommentActivity* 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 *AppController*. Dieser regelt die Anfragen (Get- und Post-Request) an den Server innerhalb einer Request Queue. Über ein Listener-Pattern werden die Klassen *MainActivity*, *PostBoardActivity*, *WriterActivity* und *CommentActivity* an den Controller gebunden. Über die Listener-Schnittstelle *onResponse* erhalten die Activitys die angeforderten Daten als Json-Objekte. Diese werden mithilfe von Gson geparkt und in Java-Objekte überführt.

5.1 Implementierung

5.1.1 Kilroy App

Die KilroyApp besteht aus mehreren Activities, die von der *MainActivity* gesteuert werden. Die Klasse *AppController* steuert die Request Queue über diese die einzelnen Activities Requests an das Backend schicken. Für das json Parsing wird Google gson [<https://sites.google.com/site/gson/Home>] genutzt.

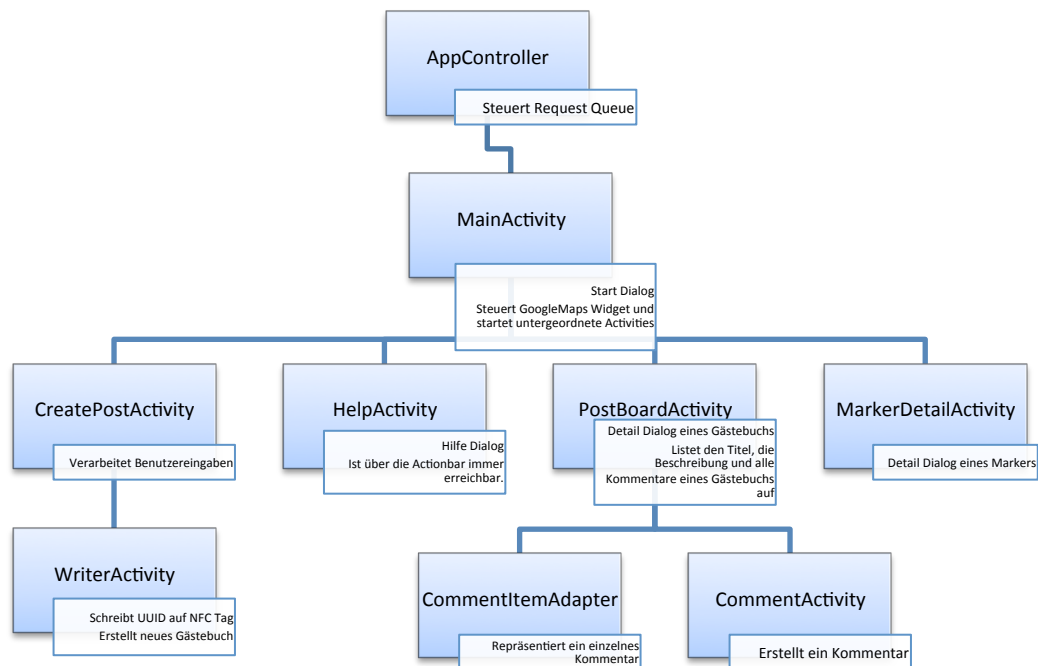


Abbildung 7: Activies

Die *CreatePostActivity* übermittelt der *WriterActivity* die Benutzereingaben. Dieser Zwischenschritt ist nötig, da sicher gestellt sein muss, dass der Benutzer mit seinen Eingaben fertig ist, um den Tag nicht frühzeitig zu beschreiben. Die *WriterActivity* aktiviert die NFC Writer Toolbox [<https://github.com/skjolber/ndef-tools-for-android>], *NfcTagWriterActivity*, welche nützliche NFC Methoden zur Verfügung stellt, indem die Parent Methode *setDetecting(true)* aufgerufen wird. Die *PostBoardActivity* wird aus der *MainActivity* gestartet. Durch die im Manifest hinterlegten Intent-Filter, wird die *MainActivity* der KilroyApp von dem Android System automatisch gestartet, sobald der Benutzer sein Smartphone an einen Tag hält. Die *MainActivity* liest den Tag aus und startet die *PostBoardActivity* mit dem richtigen Gästebuch. Entscheidet sich der Benutzer ein Kommentar auf dem Gästebuch zu hinterlassen, indem er das Erstellen

Icon drückt, wird die *CommentActivity* gestartet. Die *CommentActivity* verarbeitet die Benutzereingaben und fügt das Kommentar in das Gästebuch ein.

Einen Hilfe Dialog stellt die Activity *HelpActivity* zur Verfügung, dieser ist aus jeder anderen Activity über die ActionBar erreichbar. Klickt im Start Dialog ein Benutzer auf einen Marker doppelt, wird die *MarkerDetailActivity* gestartet. Hier wird lediglich der Titel und die Beschreibung des Tags angezeigt.

5.1.2 Kilroy Backend

Das Backend der Kilroy App besteht aus einem schlanken Webservice, der das Java Framework Spark [<http://sparkjava.com>] nutzt. Für Datenbank Anbindung an die MySQL Datenbank wird die Bibliothek sql2o [<http://www.sql2o.org>] verwendet. Außerdem wird wie in der App Google gson [<https://sites.google.com/site/gson/Home>] für das json Parsing verwendet.

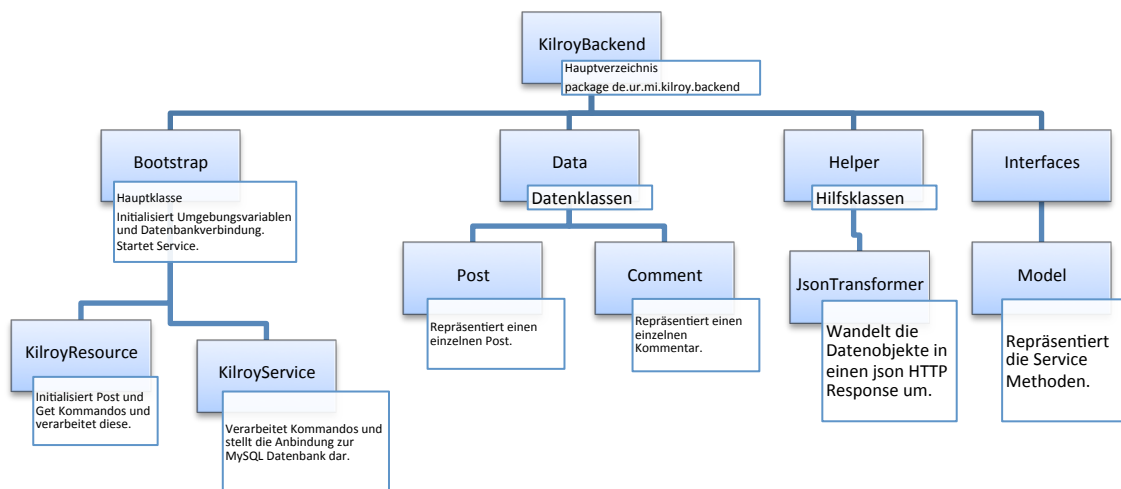


Abbildung 2: Backend Klassen

Die Klasse *Bootstrap* stellt den Einstiegspunkt des Webservices dar. In der *main*-Methode werden die Umgebungsvariablen und die Datenbankanbindung initialisiert, anschließend der Webservice gestartet. Die Klasse *KilroyResource* implementiert die verschiedenen Post- und Get-Requests und leitet diese an die Klasse *KilroyService* weiter. Die Datenbank Anfragen anschließend im *KilroyService* generiert und an die Datenbank gesendet. Der *JsonTransformer* hilft dabei, einen Response aus einer Datenklasse zu generieren.

6 Testing

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 unter anderem 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 weichen 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 *MarkerDetailActivity*, 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 *MarkerDetailActivity* 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 *MarkerDetailActivity*.

7 Finaler Zustand und Ausblick

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 Nutzerregistrierung 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.

8 Projektmanagement

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 gemeinsamen 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 gemerged.