

Neighbour Discovery in Ad-hoc Multi-hop Netzwerken

Lehrveranstaltung:

Projektstudium

Dozent:

Prof. Dr. Huhn

Mitglieder:

Tom Morelly

Peter Meindl

Rico Eisenberg

Adrian Saiz

Danil Zakablukovskiy

Inhaltsverzeichnis

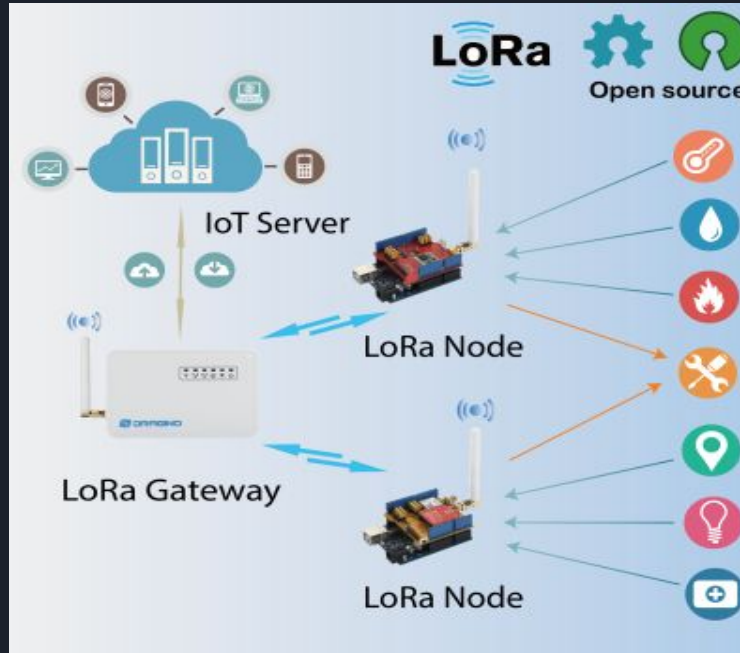
1. Einführung
 - a. LoRa
2. Projekt
 - a. Projektauftrag
3. Planung
 - a. Teams
 - b. Zeitplan
4. Umsetzung
 - a. Team: Android
 - b. Team: Neighbour Discovery
 - c. Team: Tests / QA
5. Fazit
 - a. Probleme
 - b. Ergebnis
 - c. Fazit
6. Live Demonstration



Einführung



Einführung



Projekt



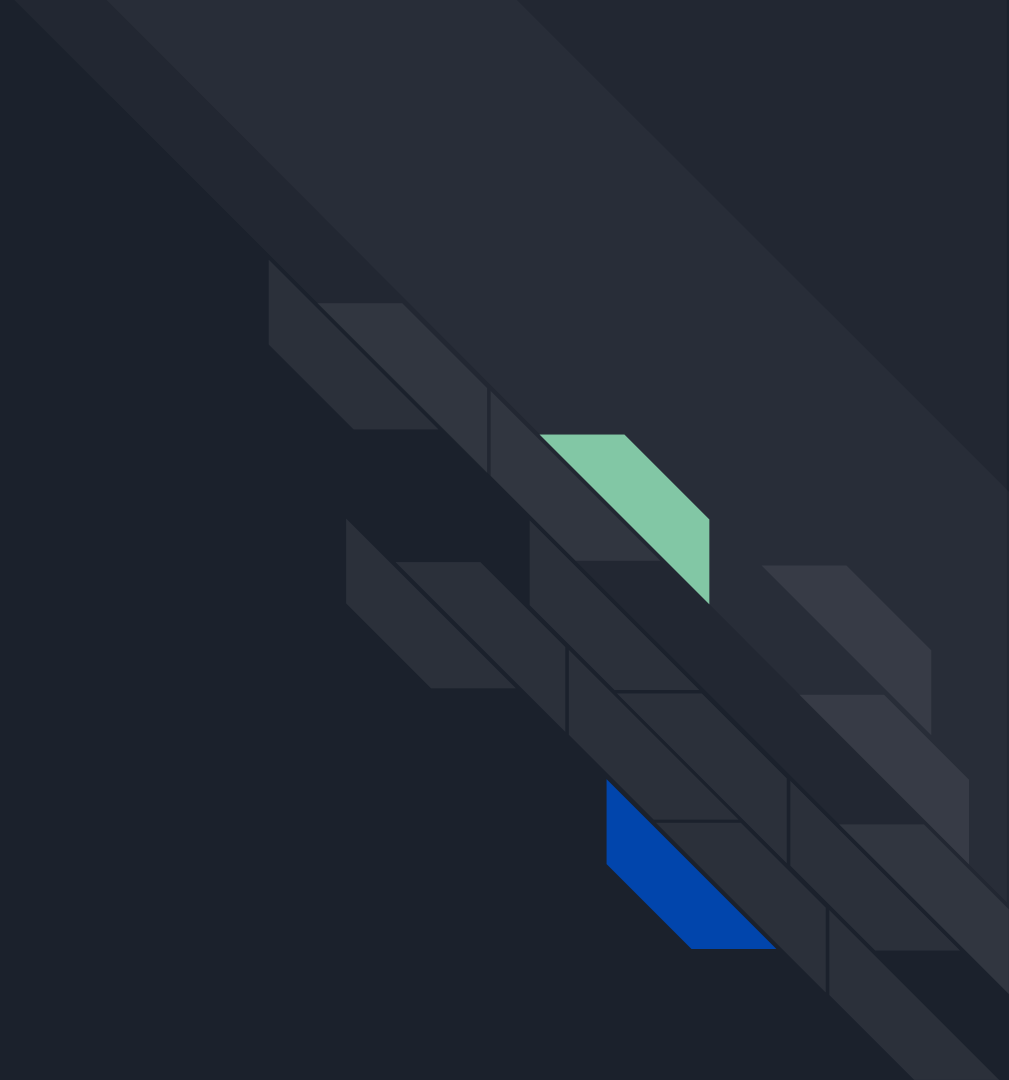


Projektauftrag

Android Applikation entwickeln, die in der Lage ist:

- Bluetooth Geräte zu finden und sich mit diesen zu Verbinden,
- über AT-Befehle mit LoRa Modulen kommuniziert,
- einen Neighbour Discovery Algorithmus implementiert,
- gefundene LoRa-Endpunkte auf einer Karte darstellt.

Planung

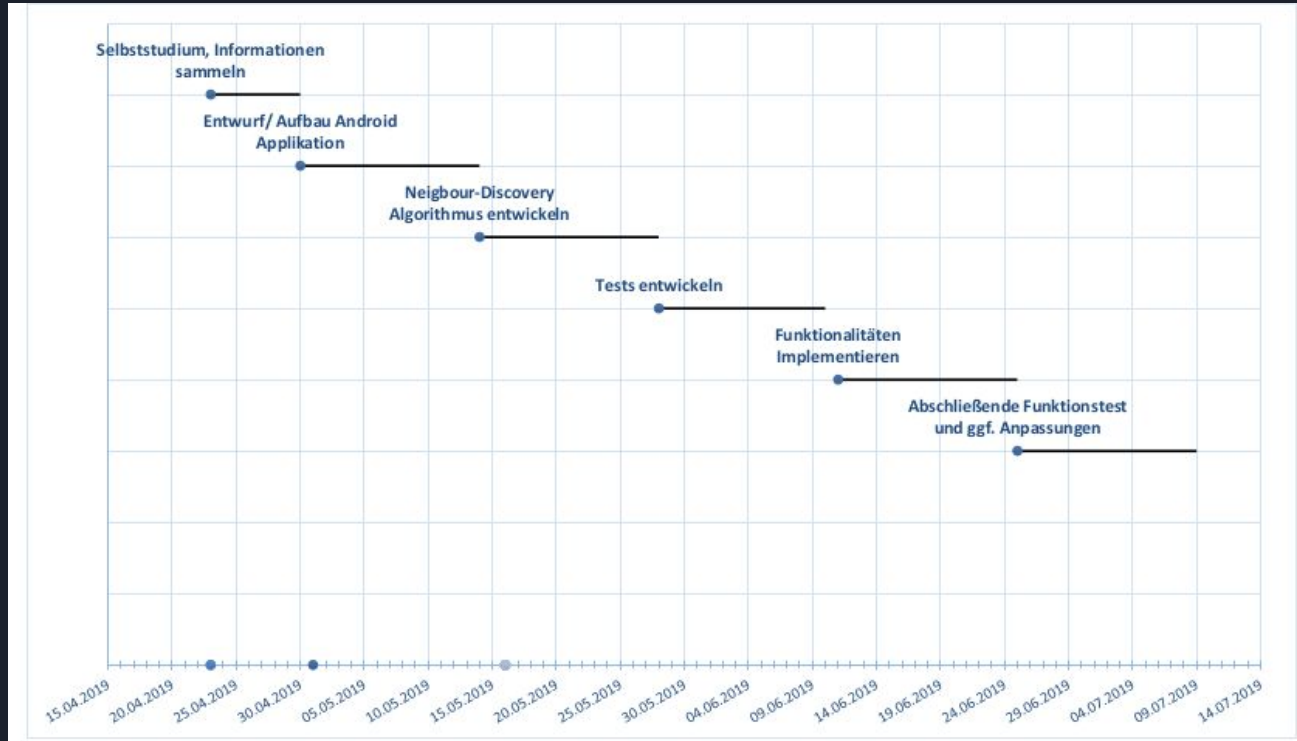




Teams

Team	Personen
Android (UI, Architektur, Bluetooth, Map)	Rico, Adrian
Neighbour Discovery (Algorithmus, Komponenten)	Tom, Peter, Danil
QA (Tests, Design / Architektur)	Danil

Zeitplan

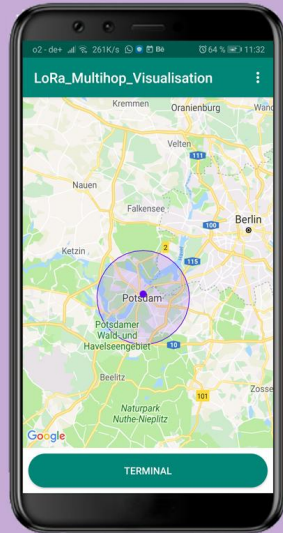


Umsetzung



vorgegebene Hardware

Überblick der Hardwarekomponenten und deren Kommunikation



Bluetooth



UART (AT Commands)



LoRa



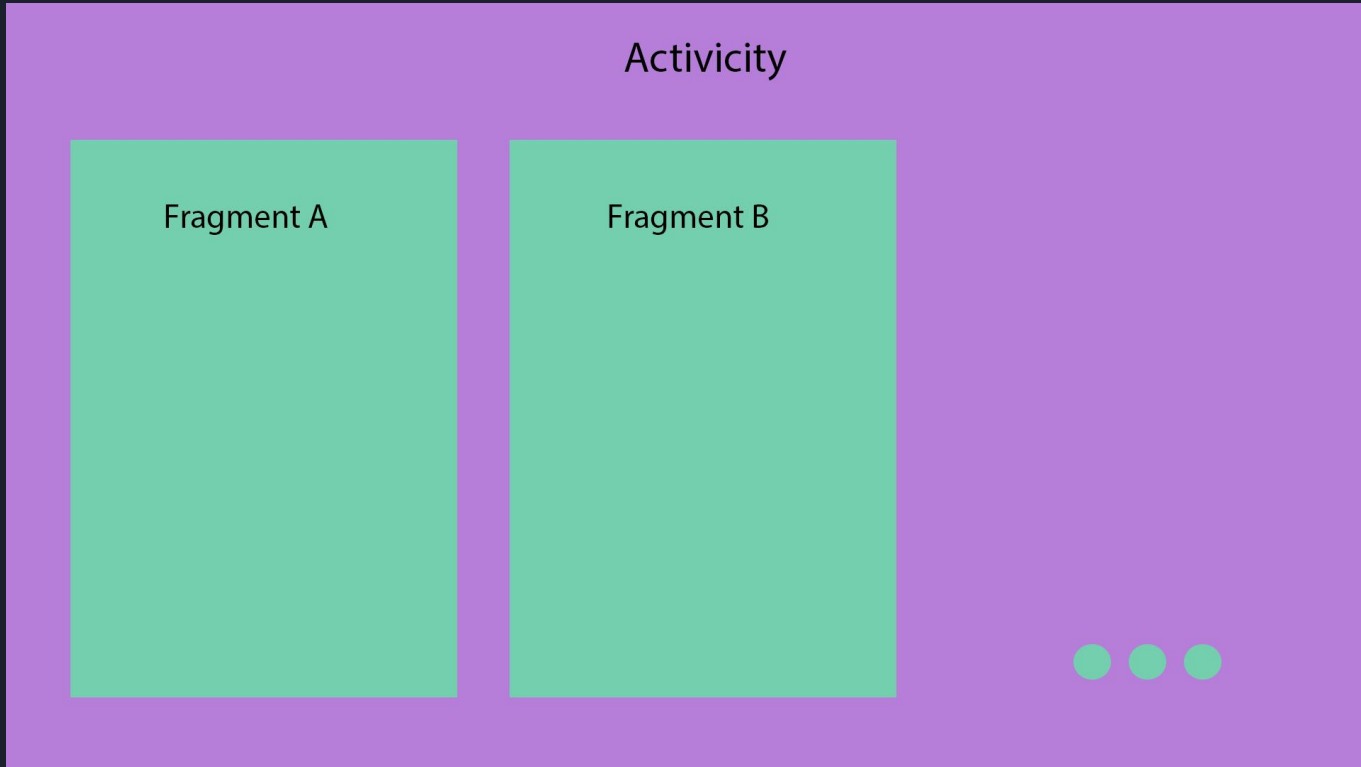
Kommunikation



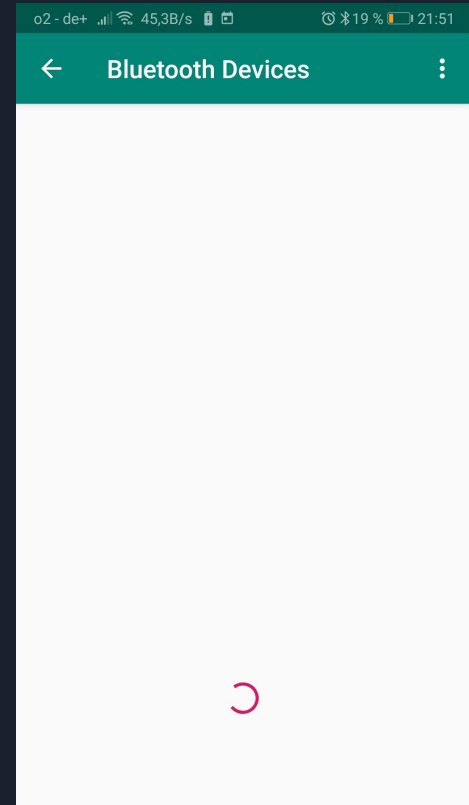
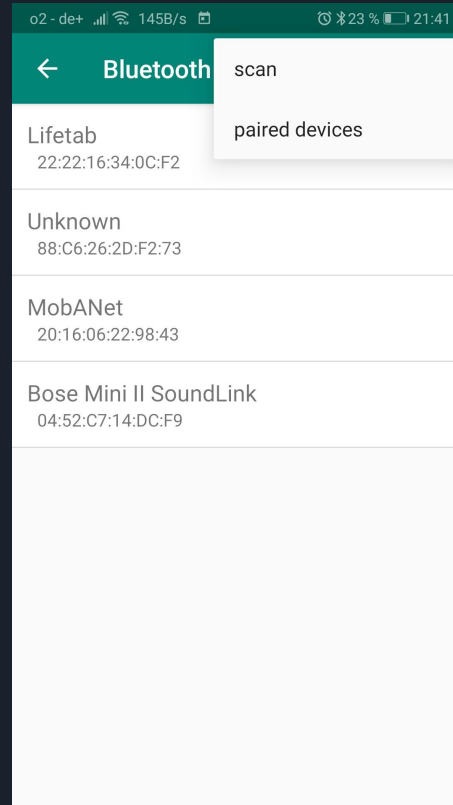
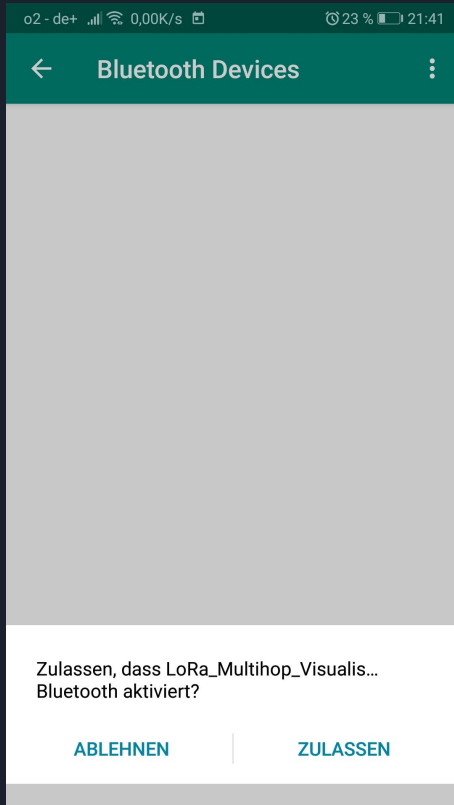
Stromversorgung



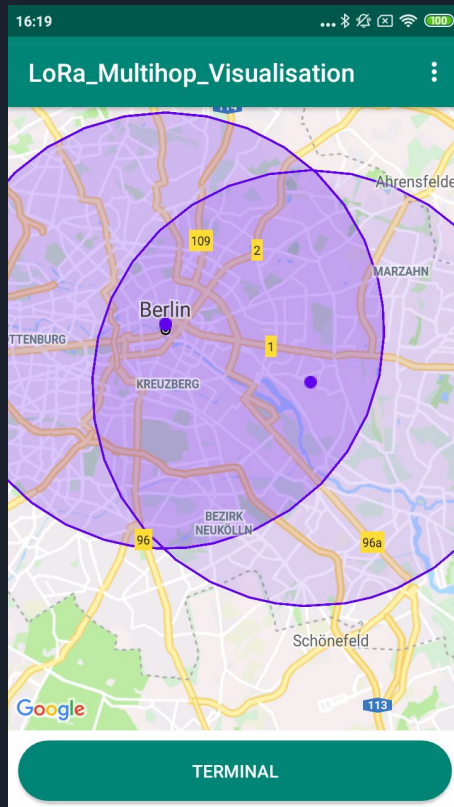
User Interfaces in Android



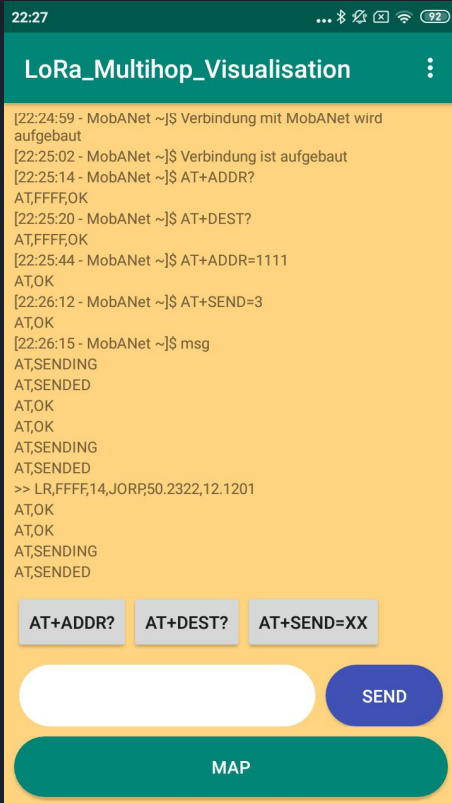
Select Device Activity



Map Fragment



Terminal Fragment



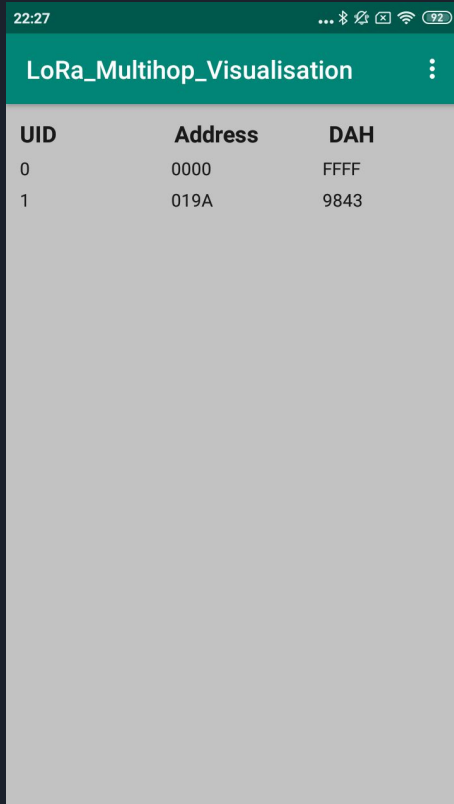
```
22:27 ... 92%
LoRa_Multihop_Visualisation
[22:24:59 - MobANet ~]$ Verbindung mit MobANet wird
aufgebaut
[22:25:02 - MobANet ~]$ Verbindung ist aufgebaut
[22:25:14 - MobANet ~]$ AT+ADDR?
AT,FFFF,OK
[22:25:20 - MobANet ~]$ AT+DEST?
AT,FFFF,OK
[22:25:44 - MobANet ~]$ AT+ADDR=1111
AT,OK
[22:26:12 - MobANet ~]$ AT+SEND=3
AT,OK
[22:26:15 - MobANet ~]$ msg
AT,SENDING
AT,SENDED
AT,OK
AT,OK
AT,SENDING
AT,SENDED
>> LR,FFFF,14,JORP,50.2322,12.1201
AT,OK
AT,OK
AT,SENDING
AT,SENDED
```

AT+ADDR? AT+DEST? AT+SEND=XX

SEND

MAP

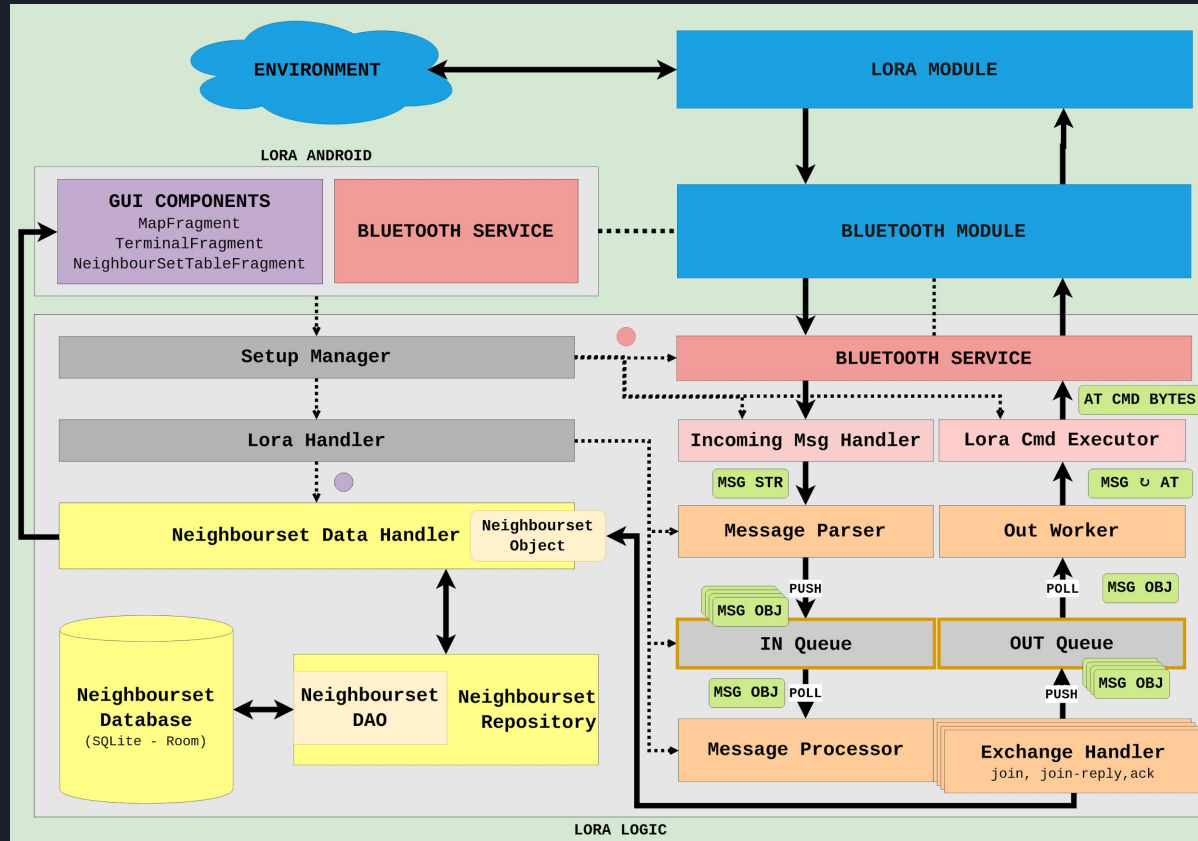
NeighbourSetTabel Fragment



The image shows a mobile application interface. At the top, there is a status bar with the time 22:27 and various system icons. Below this is a teal header bar with the text 'LoRa_Multihop_Visualisation' and a vertical ellipsis menu icon. The main content area is a table with three columns: 'UID', 'Address', and 'DAH'. The table contains two rows of data.

UID	Address	DAH
0	0000	FFFF
1	019A	9843

Systembild





Neighbour Discovery Algorithmus



Team: Test / QA

Unittests und Integrationstests (Android Instrumented tests).

Unittests:

- führen sich in Java umgebung
- schnell und leicht zu ausführen
- testen separate Klassen und Methoden
- standard JUnit framework ist benutzt

Zum Beispiel: ParseJoinMessagesTest, Parse*MessagesTest, ProcessorTest, ConfiguratorTest, NeighbourSetDatabaseTest, ...

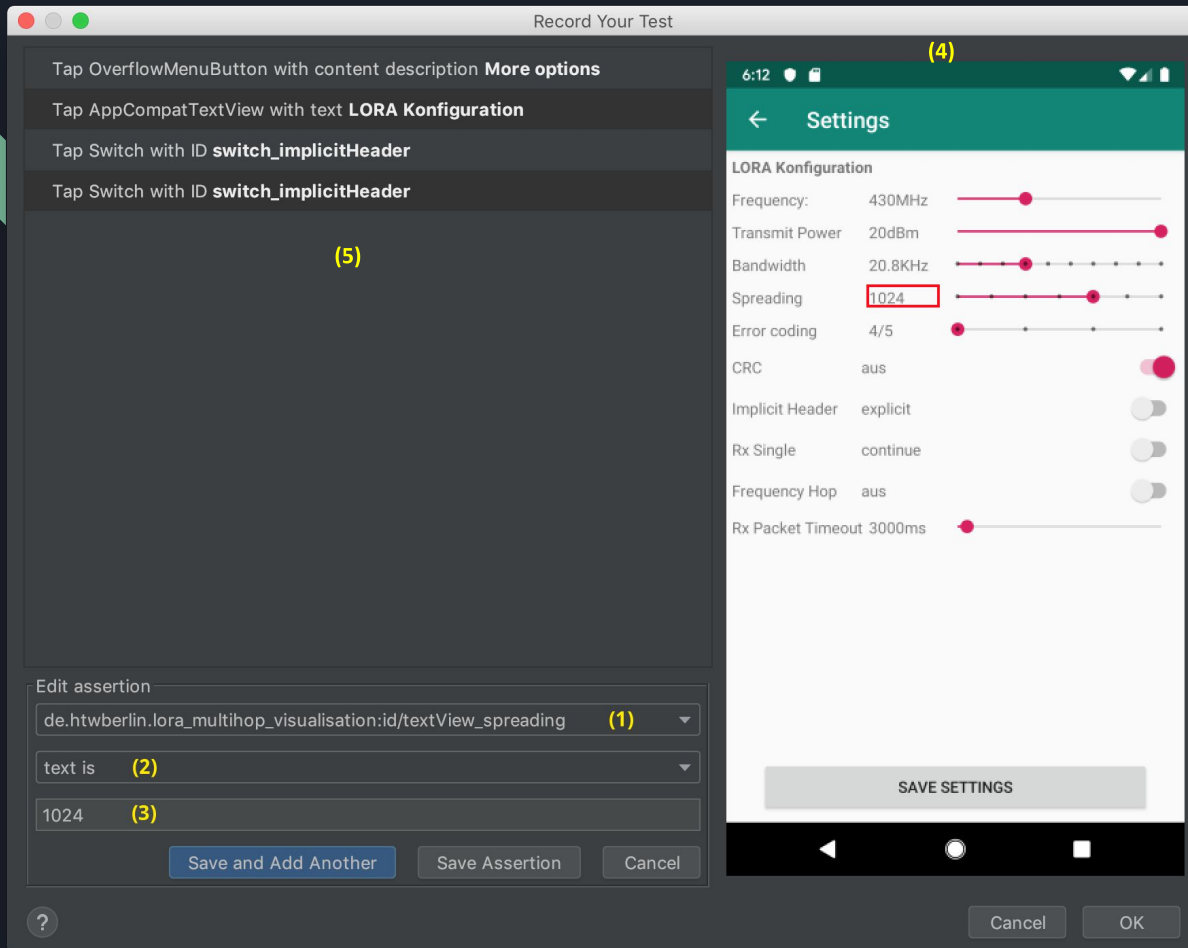


Team: Test / QA

Integrationstests:

- führen sich in einen Android Emulator aus, langsam
- testen Activities, UI und Interaktion der verschiedenen Komponenten
- kann man in Android Studio automatisch als Espresso Tests verzeichnet und generiert werden
- **Bluetooth Service** ist nicht verfügbar, deshalb gibt es keine Möglichkeit um LoRa Modul zu testen

Zum Beispiel: SettingsActivityTest, NeighbourSetTableFragmentTest, ...



(1) Active UI Element

(2) Assertion rule

(3) Assertion value

(4) Current App state

(5) Actions history

Fazit





Ergebnis

Meilenstein	Status	Kommentar
Verbindung mit Bluetooth Geräten aufbauen	done	
Kommunikation mit LoRa-Modulen mittels AT-Befehlssatz	done	
LoRa-Nachrichten zwischen Modulen senden & empfangen	done	
JOIN Mechanismus	done	
FETCH Mechanismus	implemented	not tested, due to the lack of existing modules
Pull Mechanismus	implemented	not tested, due to the lack of existing modules
Devices im Neighbour Set auf Map darstellen	done	
Testabdeckung	done	What is testable with reasonable effort is tested.
Github (Readme, Wiki, Release)	to be done	Will be completed with the official repository



Probleme

Team	Herausforderungen	Lösungen
Android	<ul style="list-style-type: none">• GPS Location variiert je nach API-Level• DB -> NeighbourSet Fragment -> Map	<ul style="list-style-type: none">• Hardcodieren• Pair Programming
Neighbour Discovery	<ul style="list-style-type: none">• OO-Design• unzuverlässige Kommunikation mit Modulen (bspw. CPU:ERR:BUSY)	<ul style="list-style-type: none">• Refactoring / Pair Programming• Timeouts / Threading
QA	<ul style="list-style-type: none">• Android Studio unterstützt kein Bluetooth	<ul style="list-style-type: none">• Hardware-Smartphones



Fazit

-



Live Demo



Fragen ?