

Freifunk Reichweiten-Erfassung und -Darstellung (Kurz F.R.E.D.)

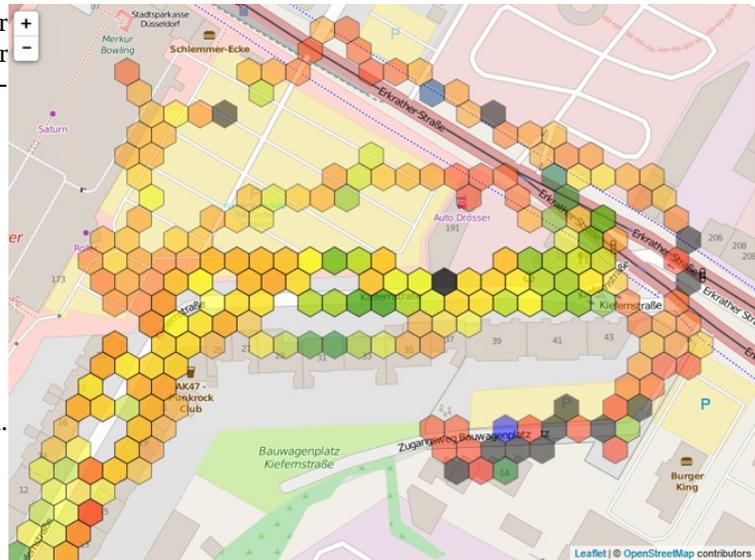
Schon im Jahr 2014 gab es [Bestrebungen](#) die Reichweite und tatsächliche Nutzbarkeit der Freifunk-Knoten grafisch auf einer Karte zu visualisieren.

Da nun im Rahmen einiger Förderprojekte viel Freifunk gerade in Innenstädten ausgebaut wurde, möchte der Förderverein freie Infrastruktur e. V. auf Anregung von Matthias Walther dieses Projekt wiederbeleben und eine marktreife App zur Datenerfassung, ein Backend zur Datenarchivierung und ein Plugin für die Freifunk-Karte MeshViewer erstellen lassen, welche mit minimalem Aufwand von jeder Freifunk-Community genutzt werden kann.

Letzteres soll eine Darstellung der Verbindungsgeschwindigkeit anhand einer Farbskala als zusätzlichen Layer in den Freifunk-Karten der Communities ermöglichen.

Das Projekt soll folgende Phasen durchlaufen:

1. Festlegung der technischen Anforderung
2. Budgetbedarf ermitteln (z. B. über Freiberuflerplattformen)
3. Mitteleintreibung durch öffentliche Förderung (beispielweise Antrag an die Staatskanzlei NRW), einer Spendenkampagne z. B. auf Betterplace oder Zuschüsse aus Freifunkvereinen. Ggfs. auch die Kombination mehrerer Finanzierungsquellen
4. Durchführungsphase: Auftragserteilung und Abnahme der Software



Phase 1: Technische Anforderungen

Das Projekt besteht aus drei technischen Software-Modulen, die nach definierten Spezifikationen miteinander interagieren. Diese sind Datenerfassung mittels Android-App, Speicherung der Daten in einer freien Datenbank und Visualisierung der Daten mittels eines Plugins im MeshViewer respektive der Freifunkkarte der Community.

Die gesamte erstellte Software muss unter der GPLv3 in einem Git-Repository übergeben und veröffentlicht werden. Vorzugsweise direkt auf Github. Die Einreichung und Genehmigung der App im Google Play Store gehört zur Vertragserfüllung dazu.

Datenerfassungs-App:

- Wegen der großen Verfügbarkeit und praktikablen, beiläufigen Datenerfassung aus der Hosentasche heraus bietet sich die Erfassung via Android-App an.
- Community-Konfiguration: Die App soll über den Play-Store heruntergeladen werden. Eine App für alle Communities. Die Konfiguration sollte idealer Weise über die Freifunk-API erfolgen, in der jede Community hinterlegen kann, wie ihre SSIDs heißen und wo die gesammelten Daten hochgeladen werden sollen.
 - Die Liste der Communities wird automatisch aus der Freifunk-API geladen und der Nutzer wählt seine Community aus einer Liste aus
 - Die App übernimmt dann automatisch die Konfiguration für die korrekte nodes.json und die Gegenstelle zum Hochladen der Daten
- Zwischenspeicherung, Hochladen nur über WLAN
- Darstellung der aktuellen Position auf einer OpenStreetMap-Karte zur Verifikation der Position
- Darstellung der Knoten über die nodes.json zur Orientierung
- Gesichert werden sollen folgende Informationen:
 - Zeitstempel
 - Position und deren Genauigkeit
 - freiwillig Name des Nutzers / Pseudonym
 - verw. Hardware / Telefonmodell
 - eigene Bewegungsgeschwindigkeit
 - genutzte Bewegungsart (zu Fuß / Fahrrad / Auto / Bus etc.)
 - Verbindungsqualität

- Freifunk-Knoten (Mac-Adresse und Namen über Statusseite)
- dBi-Wert
- WLAN-Geschwindigkeit in Mbit/s
- Optional: Speedtest per Iperf3 vom Gateway der Community

Backend:

- Freie Datenbanksoftware (bspw. MySQL oder PostgreSQL)
- Datenexport via JSON, SSL verschlüsselt
- Kompatibel zu Ubuntu 18.04 Server oder Debian 9 Server
- Periodische Berechnung der Daten für eine Kachel
 - Gewichtung sollte per Konfigurationsdatei einstellbar sein
 - Ausschließen bestimmter Benutzer (sofern erforderlich)
- Bereitstellung des Exports als JSON-Datei
 - Kacheln sollten einen Durchmesser von 5-10 m haben
 - Indizierung über Koordinaten
 - Skalenwert für mittleren Durchsatz und mittlere Verbindungsqualität

MeshViewer:

- Die Daten sollen in der Freifunkkarte der jeweiligen Community dargestellt werden können. Die verbreitetste Software ist der MeshViewer
- In Absprache mit den Maintainern als Plugin oder direkt als Vollintegration die Darstellung der Verbindungsqualität und ggfs. Durchsatztests. Der zusätzliche Layer sollte sich mit einem Klick aktivieren und deaktivieren lassen

Phase 2: Ggfs. Projekt gesamt oder die drei Teile einzeln über Freiberuflerplattformen anbieten, Angebote einholen.

Phase 3: Mittelgenerierung. Ggfs. über Förderprogramme, eine Betterplacekampagne oder aus Mitteln der Freifunkvereinen

Phase 4: Abgleich der Anforderungen mit erstellter Software