

Technická dokumentace  
GPS Tracking Systém “LOTR”

Tým LOTR

5. listopadu 2025

# Prohlášení

Prohlašujeme, že tento dokument byl vypracován samostatně a všechny použité zdroje jsou řádně citovány.

V Praze dne 5. listopadu 2025

*Tým LOTR*

# Poděkování

Děkujeme všem, kteří se podíleli na tomto projektu.

## **Abstrakt**

Tento dokument poskytuje komplexní technický přehled systému pro sledování GPS zařízení "LOTR". Popisuje architekturu backendu, hardwarových komponent a mobilní aplikace.

ableofcontents

# Seznam obrázků

# Seznam tabulek

# Listings



# Část I

## Analytická část

# Kapitola 1

## Úvod

### 1.1 Kontext a motivace

Stručný popis problému sledování polohy, využití v praxi a motivace projektu LOTR.

### 1.2 Cíle projektu

- Navrhnout a implementovat nízkoenergetický GPS tracker.
- Vytvořit backend (Node.js) s API a administračním rozhraním.
- Dodat mobilní aplikaci (Android) s funkcemi dohledu.
- Zajistit bezpečnost, škálovatelnost a spolehlivost řešení.

### 1.3 Rozsah a omezení

Co je a co není součástí tohoto projektu (např. iOS aplikace, pokročilá analytika atd.).

### 1.4 Stakeholderi a uživatelské role

Přehled rolí: uživatel, administrátor, servisní technik.

### 1.5 Pojmy a zkratky

Seznam důležitých pojmů (GNSS, OTA, ERD, REST, JWT, ...). V případě potřeby rozšiřte v příloze Seznam zkratk.

# Kapitola 2

## Systémový přehled

### 2.1 Komponenty systému

Přehled hlavních komponent: HW tracker, backend server, databáze, web/administrace, mobilní aplikace.

### 2.2 Use-cases a scénáře

Typické scénáře: registrace zařízení, odesílání polohy, dohledové zóny, upozornění.

### 2.3 Nefunkční požadavky

Dostupnost, spolehlivost, škálovatelnost, bezpečnost, spotřeba energie.

# Kapitola 3

## Architektura systému

### 3.1 Celková architektura

Vysokourovňový diagram komponent a toků dat.

### 3.2 Komunikační rozhraní

Protokoly a formáty (HTTP/HTTPS, REST, JSON, MQTT pokud relevantní).

### 3.3 Diagramy

Sekvenční diagramy pro klíčové toky: registrace, přenos polohy, OTA.

## Část II

### Návrh a implementace

# Kapitola 4

## Backend server (Node.js)

### 4.1 Přehled a technologie

Node.js, Express, nasazení, konfigurace prostředí.

### 4.2 Struktura aplikace

Adresářová struktura, hlavní moduly a jejich odpovědnosti.

### 4.3 Konfigurace a nasazení

Proměnné prostředí, build a běh, CI/CD (pokud existuje), monitoring.

# Kapitola 5

## Databáze

### 5.1 Model a ER diagram

Entity, vztahy, schéma, indexy, migrační strategie.

### 5.2 Integritní pravidla a výkonnost

FK, validace, optimalizace dotazů.

# Kapitola 6

## API a autentizace

### 6.1 Přehled REST API

Hlavní routy, verze API, konvence, chybové kódy.

### 6.2 Autentizace a autorizace

JWT, správa relací, role a oprávnění, rate limiting.



# Kapitola 7

## Zařízení a hardware

### 7.1 Popis komponent

LilyGO T-Call, GNSS modul, napájení, baterie, antény.

### 7.2 Schéma zapojení a konstrukce

Elektrické schéma, mechanická konstrukce, pouzdro.

### 7.3 Spotřeba a provozní profily

Měření, optimalizace, doporučení.

# Kapitola 8

## Firmware

### 8.1 Architektura firmware

Moduly, stavy, režimy spánku a probouzení.

### 8.2 Konfigurace a OTA

Konfigurační parametry, OTA aktualizace, servisní režim.

### 8.3 Datové formáty a protokoly

Struktura odesílaných dat, validace, zabezpečení přenosu.

# Kapitola 9

## Mobilní aplikace (Android)

### 9.1 Architektura a použité knihovny

Architektonický vzor, použitá SDK, knihovny, minimální verze Androidu.

### 9.2 Uživatelské rozhraní

Hlavní obrazovky, navigace, stavy, přístupnost.

### 9.3 Služby a integrace

Sběr polohy, komunikace se serverem, notifikace.

### 9.4 Konfigurace a nasazení

Build, signing, publikace (Google Play), prostředí.

# Kapitola 10

## Zpracování GPS dat

### 10.1 Pipeline zpracování

Příjem dat, validace, agregace, ukládání, notifikace.

### 10.2 Datové modely

Struktury pro body, trasy, geofencing.

### 10.3 Výkon a optimalizace

Dávkové zpracování, fronty (pokud jsou), indexy.

# Kapitola 11

## Frontend a administrace

### 11.1 Webové rozhraní

Mapy, vizualizace, filtrování, historie.

### 11.2 Administrace a správa uživatelů

Role, oprávnění, správa zařízení, auditní logy.

## Část III

### Kvalita, provoz a bezpečnost

# Kapitola 12

## Testování a kvalita

### 12.1 Unit a integrační testy

Testovací strategie na úrovni serveru, firmware, aplikace.

### 12.2 E2E testy a akceptační kritéria

Scénáře, metriky úspěchu, automatizace.

### 12.3 Měření výkonu a spolehlivosti

Load testy, sledování dostupnosti, SLO/SLI.

# Kapitola 13

## Nasazení a provoz

### 13.1 Provozní prostředí

Dev/test/prod, závislosti, konfigurace.

### 13.2 Build a release proces

Versioning, changelog, release artefakty.

### 13.3 Monitoring a logování

Zdraví služeb, metriky, alarmy.



# Kapitola 14

## Bezpečnost a soukromí

### 14.1 Bezpečnost komunikace a dat

TLS, šifrování, ochrana proti útokům.

### 14.2 Řízení přístupu a audit

Role, zásady hesel, logování.

### 14.3 Ochrana osobních údajů

Zásady GDPR, uchovávání dat, anonymizace.

## Část IV

### Závěr

# Kapitola 15

## Závěr a další rozvoj

### 15.1 Shrnutí výsledků

Hlavní dosažené cíle a přínosy.

### 15.2 Možnosti dalšího rozvoje

Technické i produktové směry, dluhy a prioritizace.

# Příloha A

## Příklady API volání

Ukázkové requesty a response v JSON. Vhodné vložit jako `listings` s typem `json`.

## Příloha B

### Schémata zapojení

Vložte schémata zapojení (obr.) s popisky a legendou.

# Příloha C

## Troubleshooting

Známé problémy a jejich řešení pro HW, firmware, server i aplikaci.