

System pro záznam a analýzu trasy uživatele

Martin Machala

Bakalářská práce
2023



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav informatiky a umělé inteligence

Akademický rok: 2022/2023

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Martin Machala**
Osobní číslo: **A20272**
Studijní program: **B0613A140020 Softwarové inženýrství**
Forma studia: **Prezenční**
Téma práce: **Systém pro záznam a analýzu trasy uživatele**
Téma práce anglicky: **A System for Recording and Analyzing a User's Route**

Zásady pro vypracování

1. Vypracujte literární rešerši na dané téma.
2. Navrhněte zařízení pro záznam trasy s využitím zvoleného mikropočítače.
3. Uvedený návrh hardwarově realizujte.
4. Vytvořte obslužné programové vybavení pro použitý mikropočítač.
5. Navrhněte a realizujte aplikaci pro zobrazení zaznamenaných informací na počítači.

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. Ashby, Darren. Electrical Engineering 101: Everything You Should Have Learned in School...but Probably Didn't. Newnes; 3rd edition, 2011. ISBN 978-0123860019.
2. BANZI, Massimo. Getting started with Arduino. 2nd ed. Farnham: O'Reilly, 2011. ISBN 978: 449309879.
3. CATSOULIS, John. Designing embedded hardware. 2nd ed. Sebastopol, CA: O'Reilly, 2005, xvi, 377 p. ISBN 0596007558.
4. MARGOLIS, Michael, Brian JEPSON a Nicholas Robert WELDIN. Arduino cookbook: recipes to begin, expand, and enhance your projects. Third edition. Sebastopol: O'Reilly Media, [2020]. ISBN 149190352x.
5. PINKER, Jiří. Mikroprocesory a mikropočítače. 1. vyd. Praha: BEN – technická literatura, 2004, 159 s. ISBN 80-7300-110-1.

Vedoucí bakalářské práce: **Ing. Jan Dolinay, Ph.D.**
Ústav automatizace a řídicí techniky

Datum zadání bakalářské práce: **2. prosince 2022**

Termín odevzdání bakalářské práce: **26. května 2023**



doc. Ing. Jiří Vojtěšek, Ph.D. v.r.
děkan

prof. Mgr. Roman Jašek, Ph.D., DBA v.r.
ředitel ústavu

Ve Zlíně dne 7. prosince 2022

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 18.5.2023

Martin Machala, v.r.
podpis studenta

ABSTRAKT

Cílem této práce bylo vytvořit návrh, hardwarovou realizaci a softwarové vybavení pro zařízení pro záznam trasy uživatele a vytvořit návrh a realizaci aplikace pro analýzu trasy uživatele.

Součástí práce bylo vypracování rešerše na dané téma. Ta je dělena na dvě části. První část porovnává různé typy existujících zařízení pro záznam trasy uživatele. Druhá část popisuje tři existující aplikace pro analýzu trasy uživatele.

Zařízení pro záznam trasy uživatele pomocí technologie GPS zaznamenává pozici uživatele, kterou následně ukládá na paměťovou kartu ve formě trasových souborů formátu GPX. Zařízení je ovládáno pomocí tlačítek, a využívá displej, na kterém zobrazuje grafické uživatelské rozhraní. Komponenty jsou propojeny a ovládány pomocí klonu vývojové desky Arduino Mega 2560. Obslužný software byl napsán v programovacím jazyce C++. Kód tohoto softwaru je dostupný online pod licencí MIT.

Aplikace pro analýzu trasy uživatele přijímá soubory vygenerované sledovacím zařízením. Tyto soubory jsou zpracovány a analyzovány. Získaná data jsou vypsána a vizualizována pomocí grafů. Aplikace byla napsána v programovacím jazyce Python. Kód aplikace je také dostupný online pod licencí MIT.

Klíčová slova: Arduino, C++, embedded, GPS, GPX, Python

ABSTRACT

The goal of this thesis was to create a design, a hardware implementation, and the needed software for a device for recording a user's route and to design and develop an application for analyzing a user's route.

A part of this thesis was conducting research on the given topic. Its first part compares different types of existing recording devices. The second part describes three existing applications for route analysis.

The device for recording a user's route uses GPS technology to record the position of a user, which is then saved to a memory card in the form of a GPX track file. The device is controlled using buttons and it uses a display to show a graphical user interface. The components are connected and controlled using a clone of the Arduino Mega 2560 development board. Its software was written in the programming language C++. The code of this software is available online under the MIT license.

The application for analyzing a user's route accepts files generated by the recording device. These files are processed and analyzed. The acquired data are written out and visualized using graphs. The application was written in the programming language Python. The code of this application is available online under the MIT license.

Keywords: Arduino, C++, embedded, GPS, GPX, Python

Tímto bych chtěl poděkovat vedoucímu práce panu Ing. Janu Dolinayovi, Ph.D. za odborné vedení, vstřícnost a rady při vypracování bakalářské práce.

Prohlašuji, že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD	10
I TEORETICKÁ ČÁST	11
1 TECHNOLOGIE GPS	12
1.1 SATELITNÍ KONSTELACE.....	12
1.2 PŘESNOST.....	12
1.2.1 Poloha.....	12
1.2.2 Čas.....	12
1.3 DALŠÍ POZIČNÍ SYSTÉMY	13
1.4 FORMÁT GPX	13
2 ZAŘÍZENÍ PRO ZÁZNAM TRASY UŽIVATELE	14
2.1 CHYTRÉ TELEFONY.....	14
2.2 SAMOSTATNÁ ZAŘÍZENÍ	14
3 SYSTÉMY PRO ANALÝZU TRASY UŽIVATELE	16
3.1 MAPY.CZ.....	16
3.1.1 Vizualizace uživatelských dat.....	16
3.2 GOOGLE EARTH PRO	16
3.2.1 Vizualizace uživatelských dat.....	17
3.3 GPS VISUALIZER.....	17
3.3.1 Vizualizace uživatelských dat.....	17
4 VÝVOJOVÁ PLATFOMA ARDUINO	18
4.1 VYUŽITÍ	18
4.2 DĚLENÍ HARDWARU	18
4.2.1 Vývojová deska.....	18
4.2.2 Shield.....	18
4.2.3 Modul	19
4.3 PLATFORMIO	19
II PRAKTICKÁ ČÁST	20
5 ZAŘÍZENÍ PRO ZÁZNAM TRASY UŽIVATELE	21
5.1 NÁVRH ZAŘÍZENÍ.....	21
5.2 HARDWAROVÁ REALIZACE.....	21
5.2.1 Mikropočítač	21
5.2.2 GPS modul	22
5.2.3 Čtečka SD karet.....	23
5.2.4 Displej	23
5.2.5 Tlačítka.....	24
5.2.6 Finální návrh	24
5.3 OBSLUŽNÝ SOFTWARE.....	25
5.3.1 display manager	27
5.3.1.1 init	27
5.3.1.2 draw	27
5.3.1.3 clearScreen.....	27
5.3.1.4 clearRect	27

5.3.1.5	printText.....	27
5.3.1.6	drawLine	28
5.3.1.7	drawRect	28
5.3.1.8	drawBM	28
5.3.2	GUI manager	28
5.3.2.1	init	28
5.3.2.2	splash	28
5.3.2.3	base	29
5.3.2.4	date.....	30
5.3.2.5	time	30
5.3.2.6	satCount	30
5.3.2.7	infoMessage	30
5.3.2.8	errorMessage.....	30
5.3.2.9	startScreen.....	31
5.3.2.10	trackingScreen.....	31
5.3.2.11	endScreen	31
5.3.3	bitmaps	31
5.3.3.1	satellite	31
5.3.3.2	question.....	32
5.3.3.3	exclamation.....	32
5.3.3.4	splash	33
5.3.4	GPS manager.....	33
5.3.4.1	init	34
5.3.4.2	update.....	34
5.3.4.3	fix	34
5.3.4.4	getSats.....	34
5.3.4.5	getLat	34
5.3.4.6	getLon	34
5.3.4.7	getAlt	35
5.3.4.8	getHour	35
5.3.4.9	getMinute.....	35
5.3.4.10	getSecond	35
5.3.4.11	getYear	35
5.3.4.12	getMonth	35
5.3.4.13	getDay	35
5.3.5	SD manager.....	35
5.3.5.1	init	35
5.3.5.2	openFileWrite	36
5.3.5.3	fileWrite	36
5.3.5.4	fileFlush	36
5.3.5.5	fileClose.....	36
5.3.5.6	fileExists	36
5.3.5.7	seek	36
5.3.5.8	seekBack	36
5.3.6	position logger.....	37
5.3.6.1	init	37
5.3.6.2	startLogging	37
5.3.6.3	addTrackpoint	37
5.3.6.4	stopLogging	38
5.3.7	GPX generator.....	38

5.3.7.1	generateSkeleton	38
5.3.7.2	generateTrackpoint	39
5.3.8	main	39
5.3.8.1	systemError	39
5.3.8.2	readButton	40
5.3.8.3	setup	40
5.3.8.4	loop	40
5.4	HODNOCENÍ FINÁLNÍ REALIZACE	42
6	PROGRAM PRO ANALÝZU TRASY UŽIVATELE.....	44
6.1	MAIN	44
6.2	WINDOW	44
6.2.1	__init__	44
6.2.2	fileSelect	46
6.2.3	analyzeData	46
6.2.4	eraseData	50
6.3	HODNOCENÍ FINÁLNÍ REALIZACE	50
	ZÁVĚR	52
	SEZNAM POUŽITÉ LITERATURY.....	54
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	57
	SEZNAM OBRÁZKŮ	58
	SEZNAM PŘÍLOH.....	59

ÚVOD

Technologie GPS je v současné době velmi rozšířená. Využívá se například k navigaci, dokumentaci cest, vyměřování nebo kartografii. Uživatelsky nejpřístupnější zařízení obsahující GPS přijímače jsou v současné době chytré telefony. Jejich GPS přijímače ale často nejsou příliš přesné a jejich obnovovací frekvence nemusí být pro některé případy užití dostačující. Naměřené pozice tak nemusí být dostatečně přesné, a mezery mezi měřeními mohou být příliš velké, což může způsobovat problémy při určování polohy z rychlejších vozidel. Přesnější sledovací zařízení jsou dostupná, často ale za vysoké pořizovací ceny. Koncoví uživatelé tak nemají k přesným GPS sledovacím zařízením snadný přístup.

Prvním z cílů této práce je vypracovat literární rešerši na téma „Systémy pro záznam a analýzu trasy uživatele“. Rešerše by měla porovnávat různá existující řešení sledovacích zařízení a systémů pro analýzu tras.

Dalším cílem této práce je navrhnout, sestavit a naprogramovat samostatné sledovací zařízení, které využívá technologii GPS. Zařízení by mělo být přesnější a mělo by mít vyšší obnovovací frekvenci než běžná mobilní řešení. Zároveň by mělo být levnější než srovnatelná komerční sledovací zařízení. Komponenty využití k vytvoření tohoto zařízení by měly být snadno přístupné. Návrh a kód tohoto zařízení by měly být veřejně dostupné. Návrh zařízení tak může být kýmkoliv upraven a rozšířen.

Posledním cílem této práce je vytvořit desktopovou aplikaci, která je schopná převzít data vygenerována sledovacím zařízením. Tato data aplikace následně zpracuje, vypíše a vizualizuje pomocí grafů. Řešení vizualizačního programu by mělo taktéž být veřejně dostupné a upravitelné.

I. TEORETICKÁ ČÁST

1 TECHNOLOGIE GPS

GPS je poziční systém o jehož provoz se starají Vesmírné síly Spojených států amerických. Je to vesmírná konstelace družic, které obíhají Zemi. Signály těchto družic jsou přijímány GPS přijímačem, který signál interpretuje. Z přijatých dat lze určit pozici přijímače, aktuální datum a čas. Tyto údaje lze využít například k navigaci, trasování nebo mapování. [1] [2]

1.1 Satelitní konstelace

Satelitní konstelace GPS se skládá z množství družic obíhajících kolem Země. Tyto družice obíhají přibližně ve výšce 20 200 km nad mořem. K plnému pokrytí Země GPS signálem stačí 24 družic, většinu času je ale v konstelaci z důvodu redundance družic více. V době psaní této práce kolem Země obíhá 31 funkčních GPS družic. Každá z nich Zemi oběhne dvakrát za den. [2]

1.2 Přesnost

Kvalita přijímaných dat ze systému GPS je důležitá pro určování přesných údajů o poloze přijímače a aktuálním čase.

1.2.1 Poloha

Dle závazku vlády USA je snaha o to, aby kvalita vysílaných dat určených k výpočtu polohy nespadla pod 2 metry od přijímače. Přesnost přijímaných dat často bývá vyšší. [3]

Reálná přesnost dekódovaného signálu je ale často ovlivněna různými faktory mimo systém GPS. Kvalita může být ovlivněna například výkonem přijímacího zařízení, překážkami mezi přijímacím zařízením a družicemi, odražením signálu mezi budovami, rušivými signály v okolí, nebo kombinacemi všech těchto faktorů. [3]

1.2.2 Čas

V ideálních podmínkách je rozdíl získaného času ze systému GPS a reálným časem v časové zóně UTC 30 nanosekund nebo méně. [3]

1.3 Další poziční systémy

GPS je nejrozšířenější poziční systém na světě, mimo něj ale také existují další systémy s poziční funkcionalitou. [6]

Mezi tyto systémy se řadí například:

- BeiDou Navigation Satellite System – globální systém provozován Čínskou lidovou republikou, dříve zvaný Compass [6]
- Galileo – globální systém provozován Evropskou unií [6]
- GLONASS – globální systém provozován Ruskou federací [6]
- Indian Regional Navigation Satellite System / Navigation Indian Constellation – regionální systém provozován Indickou republikou, pokrývá indické území a jeho okolí [6]
- Quasi-Zenith Satellite System – regionální systém provozován Japonskem, komplementuje systém GPS, pokrývá území východní Asie a Oceánie [6]

1.4 Formát GPX

GPX je otevřený formát založený na formátu XML. Je určen pro snadnou výměnu pozičních dat mezi uživateli a programy. Forma zápisu tohoto formátu je standardizovaná, díky tomu je zaručena kompatibilita mezi různými programy, které tento formát využívají. Díky otevřenosti tohoto formátu ho je jednoduché implementovat ve vlastním softwaru. [4] [5]

2 ZAŘÍZENÍ PRO ZÁZNAM TRASY UŽIVATELE

Tato zařízení jsou schopna za pomoci satelitního přijímače získávat poziční data. Tato data jsou kompilována do trasových souborů, které jsou následně ukládány.

Různá zařízení pro záznam trasy uživatele se od sebe mohou výrazně lišit. Jak už jejich tvarovým faktorem, tak i jejich výkonem, výdrží nebo cenou.

2.1 Chytré telefony

V současné době většina nových chytrých telefonů obsahuje potřebný hardware pro příjem pozičních dat z různých satelitních konstelací. Chytré telefony jsou díky tomu uživatelsky nejdostupnější způsob, jak poziční data získávat. Kvalita těchto dat se ale mezi různými zařízeními může výrazně lišit. Je závislá na kvalitě přijímacího čipu, kvalitě antény, typu podporovaných konstelací a softwaru telefonu. Kvalita dat se tak může lišit i mezi dvěma zařízeními se stejným pozičním čipem. Lepší z telefonů dosahují průměrné chyby v měření kolem 3-5 metrů. [7]

2.2 Samostatná zařízení

Samostatná sledovací zařízení jsou specializována pro příjem pozičních dat. Jsou často schopná získávat data z různých satelitních konstelací. Obsahují také další senzory, které jsou využívány pro přesnější určení pozice. Z těchto důvodů jsou často přesnější než chytré telefony. [8]

Díky jejich specializaci jsou také více energeticky úsporné, a vydrží v provozu déle než chytré telefony. Některé modely mají na rozdíl od nových chytrých telefonů vyměnitelné baterie. [8]

Samostatná zařízení se dělí do více cenových kategorií. Levnější modely často dosahují přesnosti vyšší než 3 metry. Profesionální modely jsou schopné určovat polohu až v řádu milimetrů. [8] [9] [10] [11]

Největší nevýhodou těchto zařízení je jejich pořizovací cena. Ceny levnějších modelů, které jsou výkonově lepší než chytré telefony, se pohybují v rozmezí 100 až 400 eur (přibližně 2500 až 9500 Kč v době psaní práce). Nejvýkonnější modely mohou cenou přesáhnout 10000 eur (přibližně 250 000 Kč v době psaní práce). [8] [10] [11]

Další nevýhoda u některých z těchto zařízení je způsob exportu naměřených dat do počítače. Pro přenos dat u těchto specifických zařízení je potřeba proprietární software, který nemusí být kompatibilní s operačním systémem uživatele. V některých případech software není volně dostupný, a je potřeba ho zakoupit samostatně. [9] [11]

3 SYSTÉMY PRO ANALÝZU TRASY UŽIVATELE

Tyto systémy akceptují soubory s pozičními údaji, které obsahují informace o zaznamenaných trasách. Po přijetí a zpracování těchto dat se informace zobrazí uživateli. Informace o přijatých datech mohou být například vypsány jako text, vizualizovány pomocí grafů, nebo zobrazeny na mapových podkladech.

Různá řešení systémů pro analýzu trasy mohou přijaté trasové údaje zpracovávat a zobrazovat různými způsoby.

3.1 Mapy.cz

Mapy.cz je služba od české společnosti Seznam.cz poskytující mapové podklady. Obsahuje jak podklady atlasové, tak letecké/satelitní. Atlasová data pro Českou republiku a Slovenskou republiku jsou upravována redaktory služby Mapy.cz. Zbytek dat je převzatý ze služby OpenStreetMap. Od určité úrovně přiblížení jsou letecké mapy pro Českou republiku a Slovenskou republiku poskytovány společností Seznam.cz. Zbytek leteckých a satelitních snímků je získáván z různých externích zdrojů. [12] [13]

3.1.1 Vizualizace uživatelských dat

Mapy.cz podporují import vlastních dat ve formátu .GPX. Po importu dat se na vybraném mapovém podkladu zobrazí náhled všech tras a bodů uložených v importovaném souboru. V listu tras je možné vidět délku každé trasy v souboru. [14]

Po přihlášení se dá trasa uložit pomocí funkce „Moje mapy“. Výšková data uložená v souboru je možné přepsat výškovými daty získanými z mapových podkladů. Pokud uložená trasa obsahuje výšková data, Mapy.cz vygenerují výškový graf trasy. [14] [15]

3.2 Google Earth Pro

Google Earth Pro je desktopová aplikace společnosti Google určena pro zobrazování různých geografických dat. Data jsou zobrazována na 3D modelu Země, který lze volně prohlížet. Základní zobrazení obsahuje letecké a satelitní snímky planety. Aplikace také obsahuje velké množství vrstev, které základní zobrazení rozšiřují. [16]

3.2.1 Vizualizace uživatelských dat

Google Earth Pro nabízí nástroje pro import a vytváření vlastních geografických dat. Program dokáže pro importování dat zpracovávat jak otevřený datový formát GPX, tak velké množství proprietárních datových formátů. [16] [17]

Importovaná data jsou v aplikaci roztřízená do složek. Po importování je možné data zobrazit jako linii na modelu Země. Aplikace umí zpracovávat a zobrazovat různé údaje ze senzorů, které jsou uloženy v importovaném souboru. [17]

3.3 GPS Visualizer

GPS Visualizer je veřejně přístupný webový nástroj určený pro vizualizaci geografických dat, konverzi mezi velkým množstvím různých geografických datových formátů a provádění výpočtů s geografickými jednotkami. [18] [19]

3.3.1 Vizualizace uživatelských dat

Trasová data je možné zobrazovat nad mapovými podklady. Nástroj obsahuje množství možností pro přizpůsobení výsledné vygenerované mapy, včetně možnosti barevného vyobrazení vybraného datového bodu, například výšky, rychlosti a podobně. [20]

Nástroj také poskytuje možnost vizualizovat kombinace různých datových bodů v grafech. Jako osy je možné dosadit vybrané datové body z importovaného souboru, jako například čas, výšku nebo počet satelitů v dohledu. Na grafových datech je také možné zvolit barevné vyobrazení vybraného datového bodu, tímto grafy mohou vyobrazovat až tři dimenze dat najednou. [21]

4 VÝVOJOVÁ PLATFOMA ARDUINO

Arduino je otevřená platforma určena k vytváření řešení, která pomocí vývojových desek s mikrokontrolery kombinují hardware a software. [22]

4.1 Využití

Platforma Arduino je díky její jednoduchosti, flexibilitě a přístupnosti využívána v mnoha různých projektech. Využívá se pro hobby projekty určené k automatizaci různých úkolů, jako učební pomůcka ve školách pro práci se softwarem a elektronickými součástkami, jako cenově dostupný způsob vytváření vědeckých nástrojů nebo jako sada nástrojů pro prototypování profesionálních řešení. [22]

4.2 Dělení hardwaru

Současný Arduino hardware se dělí na tři hlavní kategorie dle jejich funkcionality.

4.2.1 Vývojová deska

Vývojové desky Arduino jsou řízené mikrokontrolerem. Mikrokontroler je programován propojením desky a počítače přes rozhraní USB. Signály získané z počítače jsou převedeny na sériový signál, který se využívá pro komunikaci s mikrokontrolerem. Vývojové desky také obsahují kolekci analogových a digitálních GPIO pinů, určených k propojování desek s různými elektronickými komponenty. Desky také pomocí pinů mohou poskytovat napájení pro elektrické obvody. [24]

Mikrokontrolery použité v deskách Arduino jsou programované v jazyce založeném na jazycích C a C++. Jazyk obsahuje sadu funkcí určených pro zjednodušení práce s vývojovými deskami Arduino. Software pro programování vývojových desek Arduino obsahuje správce knihoven, který poskytuje velké množství knihoven pro různorodý hardware, který jde k deskám připojit. [24]

4.2.2 Shield

Shieldy (štíty) jsou přídavné desky, které se napojují přímo na vývojové desky Arduino. Obsahují extra hardwarové vybavení, které rozšiřuje vlastnosti základních vývojových desek. Jeden shield může obsahovat více komponentů. Společně s shieldy se také často dodávají podpůrné knihovny určené ke zjednodušení práce s těmito shieldy. Často je možné kombinovat více shieldů zároveň. [25]

Ne všechny modely vývojových desek Arduino jsou vhodné pro využití s shieldy. Menší z oficiálních modelů desek Arduino nemají žádné oficiální přídatné desky. [23]

4.2.3 Modul

Moduly, podobně jako shieldy, obsahují extra hardwarové vybavení určeno k rozšíření funkcionality vývojových desek Arduino. Obvykle ale bývají jednodušší, a obsahují méně komponentů než shieldy. Je časté, že modul obsahuje jenom jeden hlavní komponent určen pro vývojovou desku, například senzor nebo hardwarový výstup. [25]

4.3 PlatformIO

PlatformIO je profesionální open-source nástroj určený pro vývoj softwaru pro embedded zařízení. Nástroj PlatformIO zjednodušuje a sjednocuje vývojový proces pro embedded software napříč velkým množstvím embedded platform. Nabízí nástroje určené k sestavování, debugingu a testování vyvíjeného softwaru. Dále také nabízí systém pro správu knihovnic závislostí. Nástroj lze využívat jako samostatné IDE, nebo ve formě integrace do jiného IDE. PlatformIO je kompatibilní s vývojovými deskami platformy Arduino. [26]

II. PRAKTICKÁ ČÁST

5 ZAŘÍZENÍ PRO ZÁZNAM TRASY UŽIVATELE

Zařízení pro záznam trasy uživatele slouží ke sledování pozice uživatele a generování trasových souborů formátu GPX, ve kterých jsou informace o sledování uživatele uloženy.

5.1 Návrh zařízení

Zařízení by mělo pro svoji funkci využívat mikropočítač. Mělo by být schopné pomocí GPS přijímače získávat poziční data, která jsou následně převáděna na trasové soubory formátu GPX. Tyto soubory by měly být ukládány na paměťovou kartu, a v době trasování by měly být periodicky aktualizovány novými údaji.

Zařízení by mělo být uživatelsky interaktivní. Uživatel by měl mít možnost trasování zapínat, pozastavovat i vypínat pomocí tlačítek. Uživatel by také měl mít vizuální odezvu zprostředkovanou displejem. Na tomto displeji by se mělo zobrazovat interaktivní uživatelské prostředí, které je pro uživatele jednoduché a srozumitelné.

Vybraný hardware by měl být cenově dostupný, ale dostatečně výkonný k uspokojení požadavků na rychlost a přesnost.

K realizaci požadovaných funkcí zařízení by měl být využitý obslužný software, který bude pracovat se vstupními a výstupními zařízeními připojenými k mikropočítači.

Návrh zařízení i zdrojový kód obslužného softwaru by měl být veřejně publikován, s možnostmi vlastní realizace návrhu a jeho úpravy.

5.2 Hardwarová realizace

Vybrané komponenty pro hardwarovou realizaci navrženého zařízení.

5.2.1 Mikropočítač

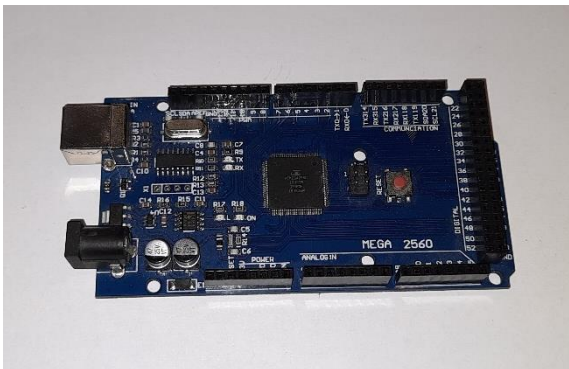
Jako mikropočítač byl vybrán klon vývojové desky Arduino Mega 2560. Platforma Arduino byla vybrána zejména díky jednoduchosti práce s jejím hardwarem a velkému množství volně dostupných knihoven pro práci s různými komponenty. Vývojové desky platformy Arduino mají otevřený design, na trhu se díky tomu objevují cenově dostupné klony oficiálních desek.

Vývojové desky Arduino lze napájet různými typy baterií, i přes 5V napájení kabelem typu USB. Jsou proto ideální pro vytváření přenosných zařízení i pro zařízení do auta.

Samotný model Arduino Mega 2560 byl vybrán zejména díky jeho velkému množství komunikačních pinů a GPIO pinů.

Po úpravě softwarových definicí může být návrh kompatibilní i s dalšími vývojovými deskami Arduino, pokud mají dostatečný počet potřebných pinů.

Vývojová deska propojuje a řídí všechny připojené komponenty pomocí obslužného softwaru, který je na ni nahraný.



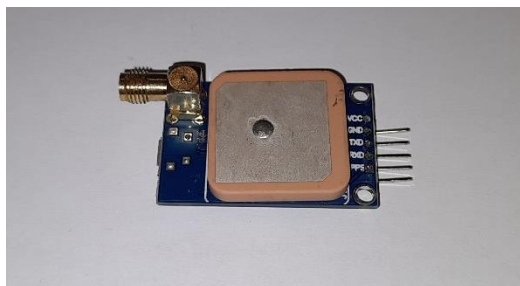
Obrázek 1: Vybraná vývojová deska Arduino Mega 2560 (klon)

5.2.2 GPS modul

GPS modul s čipem u-blox NEO-7M. Tento čip byl vybrán zejména kvůli jeho přesnosti, která je specifikována na 2.0 až 2.5 metru. Dále byl také vybrán kvůli jeho vysoké obnovovací frekvenci, která je specifikována na 10 Hz. V projektu je ale z ohledu na úložný prostor používána pouze obnovovací frekvence 1 Hz. [27]

Modul s deskou Arduino komunikuje pomocí sériové linky.

GPS modul komunikuje s GPS družicemi. Z přijatých dat počítá svoji pozici. Data jsou sbírána a zpracována mikropočítačem, ke kterému je modul připojený.



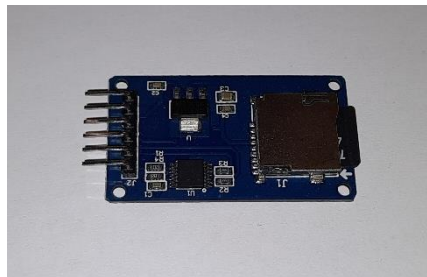
Obrázek 2: Vybraný GPS modul u-blox NEO-7M

5.2.3 Čtečka SD karet

Zařízení využívá pro čtení SD karet generický šestipinový SD modul. Vybraný modul přijímá paměťové karty typu MicroSD.

Modul se k desce Arduino připojuje pomocí rozhraní SPI.

SD čtečka je určena pro zápis trasových dat, které jsou získávány z GPS modulu, a kompilovány do trasových souborů.



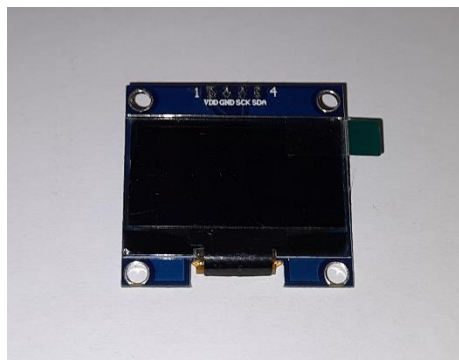
Obrázek 3: Vybraný SD modul

5.2.4 Displej

Pro tento projekt byl použitý jednobarevný OLED displej typu SH1106. Displej má rozlišení 128 x 64 pixelů.

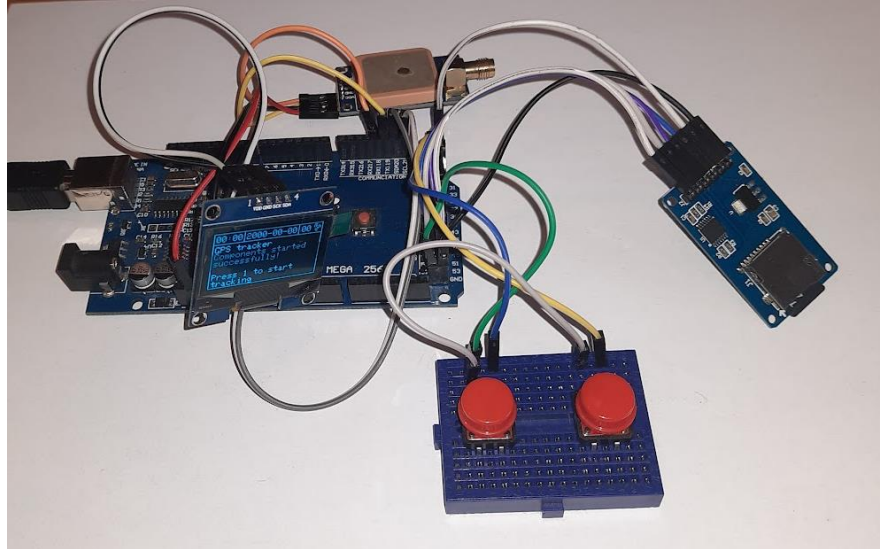
Displej je k desce připojený pomocí rozhraní I2C.

Displej slouží ke zobrazování informací o zařízení, poskytuje uživateli vizuální odezvu ohledně ovládání zařízení a vypisuje informace o aktivní sledovací relaci, vypsané informace zahrnují například čas, který od začátku uběhnul nebo současnou pozici získanou z GPS modulu.



Obrázek 4: Vybraný modul OLED displej SH1106

Ve finální realizaci jsou komponenty s deskou propojeny pomocí jumper kabelů. Tlačítka jsou napojena na nepájivé pole, které je spojeno s deskou. Fyzické zapojení je znázorněno na obrázku 7

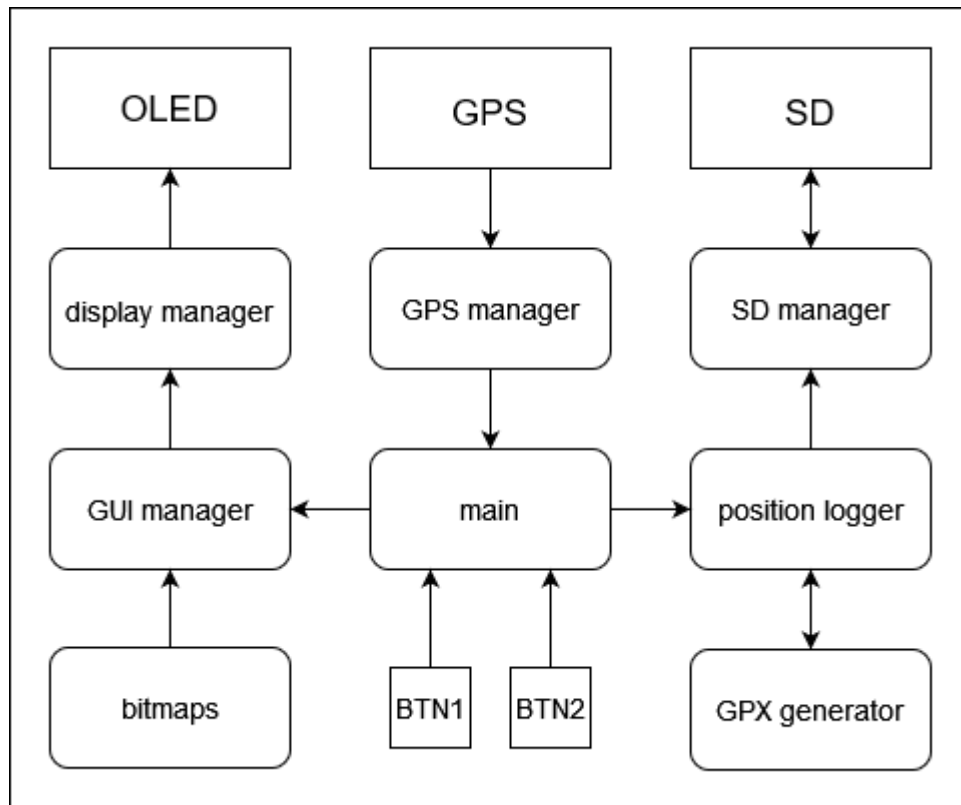


Obrázek 7: Hardwarová realizace

5.3 Obslužný software

Obslužný software zařízení pro záznam trasy byl vytvořen ve vývojovém prostředí Visual Studio Code, do kterého bylo doinstalováno rozšíření pro integraci nástroje PlatformIO. Pro jeho vývoj byl využitý jazyk C++.

Obslužný software se dělí na osm hlavních softwarových modulů, které interagují mezi sebou a s fyzickými moduly připojenými k desce. Datový tok mezi moduly je znázorněn na obrázku 8.



Obrázek 8: Blokové schéma fyzických modulů (obdélníky), softwarových modulů (zaoblené obdélníky) a datového toku mezi moduly (šipky)

Každý softwarový modul, mimo moduly *bitmaps* a *main*, obsahuje sadu funkcí určených pro práci s tímto modulem. Tyto moduly se dělí na dva soubory. První ze souborů je hlavičkový soubor formátu *.hpp*, který je určený k definování jmenného prostoru, proměnných a funkcí. Obsahuje informace, které jsou předány, když se modul zahrnuje do kódu. Druhý ze souborů je soubor typu *.cpp*, který obsahuje implementace funkcí definovaných v hlavičkovém souboru.

Modul *main* hlavičkový soubor neobsahuje, protože tento modul není zahrnován v žádném dalším modulu.

Modul *bitmaps* neobsahuje soubor typu *.cpp*, modul totiž neobsahuje žádné funkce, které je potřeba implementovat.

Moduly se dělí na tři hlavní „pilíře“. Každý z těchto pilířů slouží k obsluze jednoho hardwarového modulu. Všechny tyto pilíře jsou spojeny v hlavním modulu *main*.

5.3.1 display manager

Softwarový modul *display manager* slouží k ovládání hardwarového modulu OLED displeje. Modul ve svých funkcích implementuje funkce z knihovny Adafruit_SH110X, která je určena k ovládání OLED displejů řady SH110. Modul obsahuje funkce pro základní práci s grafikou. *Display manager* zjednodušuje práci s využitou knihovnou, a omezuje její funkcionalitu na funkce potřebné pro sledovací zařízení. Modul využívá jmenný prostor „DISP_MAN“.

5.3.1.1 init

Funkce *init* slouží k zavolání inicializační funkce displeje, ve které se nastaví správná I2C adresa. Funkce dále nastavuje velikost a barvu textu, který se na displeji bude zobrazovat. Jako poslední zobrazí výchozí úvodní obrazovku displeje. Funkce jenom volá korespondující funkci ve třídě pro ovládání displeje.

5.3.1.2 draw

Funkce *draw* vykreslí změny na displeji. Pokud funkce *draw* není zavolána, displej se neaktualizuje. To umožňuje provádět více vykreslovacích operací po sobě, a displej aktualizovat až nakonec. Funkce jenom volá korespondující funkci z knihovny pro ovládání displeje.

5.3.1.3 clearScreen

Funkce *clearScreen* vyplní celou obrazovku černými pixely. Volá se jenom korespondující funkce z knihovny pro ovládání displeje.

5.3.1.4 clearRect

Funkce *clearRect* vyplní vybranou obdélníkovou oblast displeje černými pixely. Funkce bere jako vstup pozici obdélníku a jeho výšku a šířku. Poté kontroluje, jestli jsou výška a šířka v limitech displeje. Pokud nejsou, přesune obdélník do limitů displeje.

5.3.1.5 printText

Funkce *printText* vypíše text na určitém místě obrazovky. Jako vstup funkce bere text, a jeho začáteční souřadnice. Pokud je vstupní pozice textu mimo limity displeje, je zde text přesunutý. Funkce nejdříve nastaví pozici, kde se text začne vypisovat. Poté je text znak po znaku vypisován. Pokud funkce ve vstupním textu narazí na znak nového řádku, začne vypisovat o řádek níže pod začátkem textu.

5.3.1.6 *drawLine*

Funkce *drawLine* vykreslí na displeji čáru se šířkou jednoho pixelu. Jako vstup funkce bere souřadnice začátku a konce čáry. Pokud jsou tyto údaje mimo limity displeje, jsou tam přesunuty. Čára je vykreslena světlou barvou.

5.3.1.7 *drawRect*

Funkce *drawRect* vykreslí na displeji okraje vybrané obdélníkové oblasti světlou barvou. Jako vstupní údaje funkce bere souřadnice obdélníku, jeho výšku a šířku. Pokud je pozice obdélníku mimo limity displeje, je tam přesunut.

5.3.1.8 *drawBM*

Funkce *drawBM* vykresluje na displeji bitmapu. Funkce přijímá souřadnice bitmapy, její výšku a šířku. Dále přijímá pole typu `uint8_t`. Každá položka ve vstupním poli slouží jako skupina osmi horizontálních pixelů. Bitmapa je vykreslena pozitivně, tj. nulový bit se vykreslí jako černá barva a jedničkový bit jako barva světlá.

5.3.2 GUI manager

Softwarový modul *GUI manager* využívá funkce softwarového modulu *display manager* pro vykreslování uživatelského prostředí na hardwarovém modulu OLED displeje. *GUI manager* obsahuje funkce, které slouží ke strukturování uživatelského prostředí a výpisu dat ohledně sledování, času i data. Softwarový modul také využívá softwarového modulu *bitmaps* k získávání obrázků, které jsou využívány v uživatelském prostředí. Všechny funkce modulu *GUI manager* využívají různé kombinace funkcí z modulu *display manager*. Softwarový modul využívá jmenný prostor „GUI_MAN“.

5.3.2.1 *init*

Funkce *init* zavolá inicializační funkci z modulu *display manager*. Slouží k inicializaci a nastavení OLED displeje.

5.3.2.2 *splash*

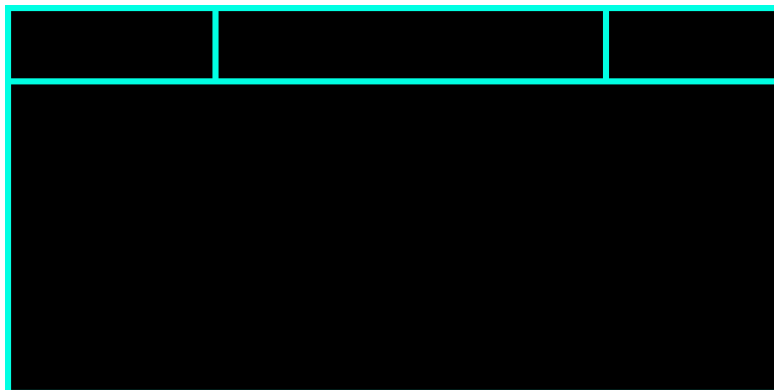
Funkce *splash* slouží k vykreslení úvodní obrazovky zařízení. Funkce vyplní celý displej černou barvou, ohraničí displej světlou barvou, vykreslí logo projektu a vypíše jméno autora. Všechny změny poté vykreslí.



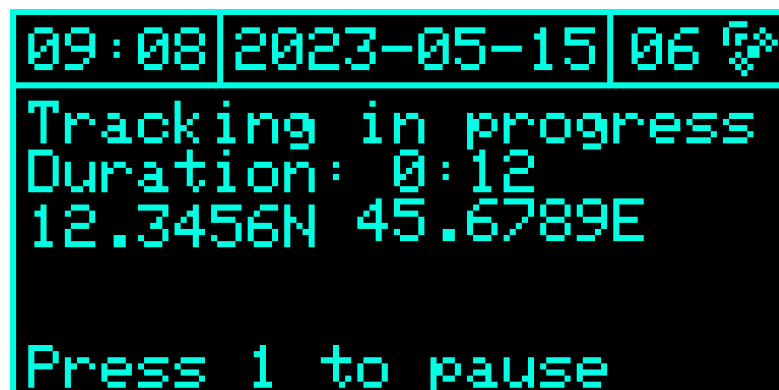
Obrázek 9: Úvodní obrazovka zařízení

5.3.2.3 *base*

Funkce *base* vykresluje rozložení hlavní obrazovky zařízení. Rozložení je na displeji vykresleno pomocí obdélníků a čar. Vytvořené oblasti jsou připraveny pro výpis informací.



Obrázek 10: Základní rozložení hlavní obrazovky



Obrázek 11: Obrazovky vyplněná daty

5.3.2.4 *date*

Funkce *date* slouží k vypsání současného data do střední horní zóny na hlavní obrazovce displeje. Funkce bere jako vstupy současný rok, měsíc a den. Vstupní hodnoty by měly mít velikost dvě číslice, pokud tomu tak není, využije se zbytek po celočíselném dělení číslem sto. Vstupní čísla jsou převedena na text ve formátu 20YY-MM-DD. Fyzický GPS modul poskytuje rok pouze jako dvě číslice, pro korektní zobrazení je před rok potřeba přidat číslo dvacet. Nejdříve je střední horní zóna vyplněna černými pixely, poté se do ní vepíše text ve správném formátu. Vše se nakonec na displeji vykreslí.

5.3.2.5 *time*

Funkce *time* slouží k vypsání současného času do levé horní zóny na hlavní obrazovce displeje. Funkce bere jako vstupy současnou hodinu a minutu. Na vstupní hodnoty je využívána funkce k získání zbytku po celočíselném dělení číslem sto, jejich velikost se tak omezí na dvě číslice. Vstupní čísla jsou převedena na text ve formátu HH:MM. Nejdříve je levá horní zóna vyplněna černými pixely, poté je do ní vložen text ve správném formátu. Displej je poté aktualizován.

5.3.2.6 *satCount*

Funkce *satCount* vypisuje do pravé horní zóny současný počet satelitů, od kterých fyzický GPS modul přijímá data. Jako vstup bere počet satelitů, z kterého je získán zbytek po celočíselném dělení číslem sto. Pravý horní roh je vyplněný černými pixely, poté je do levé části této zóny vypsán počet satelitů. Do pravé části zóny je vykreslena bitmapa v podobě GPS družice. Displej je následně aktualizován.

5.3.2.7 *infoMessage*

Funkce *infoMessage* slouží k výpisu zprávy s informací pro uživatele. Funkce vyčistí hlavní zónu obrazovky. Poté jsou v okrajích hlavní zóny vykresleny čtyři bitmapy v podobě otazníků. Nakonec je mezi otazníky vypsán text získaný jako vstup. Obraz je vykreslený na displeji.

5.3.2.8 *errorMessage*

Funkce *errorMessage* slouží k výpisu varovné zprávy pro uživatele. Funkce vyčistí hlavní zónu obrazovky. Poté se v okrajích hlavní zóny vykreslí čtyři bitmapy v podobě vykřičníků. Mezi vykřičníky je vypsán vstupní text. Obraz se vykreslí na displeji.

5.3.2.9 *startScreen*

Funkce *startScreen* zobrazí uvítací obrazovku zařízení. Nejdříve vyčistí hlavní zónu displeje, poté je vypsána uvítací zpráva s informacemi o úspěšném spuštění zařízení. Obrazovka také obsahuje instrukce ke spuštění sledování. Změny jsou vykresleny na obrazovce.

5.3.2.10 *trackingScreen*

Funkce *trackingScreen* vypisuje do hlavní zóny displeje informace o probíhající sledovací relaci sledovacího zařízení. Jako vstup přijímá počet hodin a minut strávených sledováním, zeměpisnou šířku a zeměpisnou délku. Také přijímá hodnotu, která ukazuje, zda sledování probíhá, nebo je pozastavené. Funkce nejdříve vyplní hlavní zónu černými pixely, poté vypíše zeměpisné a časové údaje. Při aktivním sledování jsou pod údaji vypsány instrukce k pozastavení sledování. Při pozastaveném sledování jsou vypsány instrukce k obnovení nebo zrušení relace. Změny jsou poté vykresleny na displeji.

5.3.2.11 *endScreen*

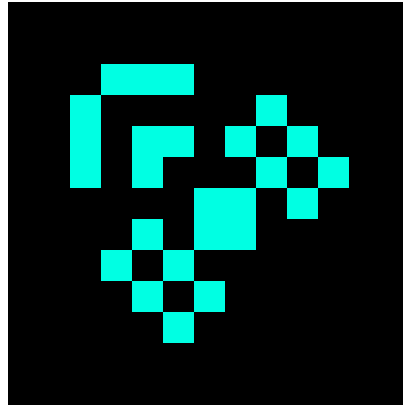
Funkce *endScreen* vypisuje do hlavní zóny displeje informace o ukončené sledovací relaci sledovacího zařízení. Jako vstupy funkce přijímá počet hodin a minut strávených ve sledovací relaci. Funkce nejdříve vyplní hlavní zónu černými pixely, poté vypíše časové údaje a instrukce k návratu na úvodní obrazovku. Změny jsou vykresleny na displeji.

5.3.3 *bitmaps*

Softwarový modul *bitmaps* obsahuje sadu jednobarevných bitmap, které jsou využívány v grafickém uživatelském prostředí. Bitmapy jsou uloženy jako pole čísel typu `uint8_t`. Ke každé bitmapě také přísluší informace o její šířce a výšce. Modul neobsahuje žádné funkce, nepotřebuje proto soubor typu `.cpp`, kde by funkce byly implementovány. Modul *bitmaps* využívá jmenný prostor „BM“.

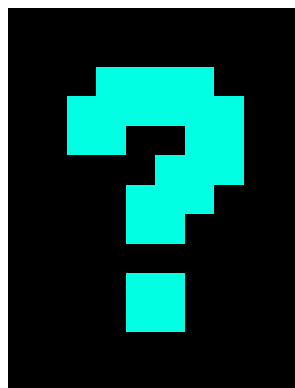
5.3.3.1 *satellite*

Bitmapa *satellite* reprezentuje satelitní družici vysílající signál. Její velikost je 9 pixelů na výšku i na šířku. Bitmapa je využívána u počítačového GPS družic, které jsou přijímány fyzickým GPS modulem.

Obrázek 12: Bitmapa *satellite*

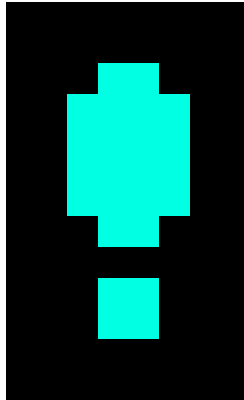
5.3.3.2 *question*

Bitmapa *question* reprezentuje otazník. Její velikost je 9 pixelů na výšku a 6 pixelů na šířku. Bitmapa je využívána při zobrazování informačních zpráv pro uživatele.

Obrázek 13: Bitmapa *question*

5.3.3.3 *exclamation*

Bitmapa *exclamation* reprezentuje vykřičník. Její velikost je 9 pixelů na výšku a 6 pixelů na šířku. Bitmapa je využívána při zobrazování varovných zpráv pro uživatele.

Obrázek 14: Bitmapa *exclamation*

5.3.3.4 *splash*

Bitmapa *splash* reprezentuje jméno projektu. Její velikost je 20 pixelů na výšku a 41 pixelů na šířku. Bitmapa je využívána v úvodní obrazovce zařízení.

Obrázek 15: Bitmapa *splash*

5.3.4 GPS manager

Softwarový modul *GPS manager* slouží k práci s hardwarovým GPS modulem. Pro interakci s hardwarovým modulem využívá třídu `Adafruit_GPS`, která slouží pro práci s množstvím kompatibilních GPS modulů. Tento softwarový modul obsahuje funkce určené převážně k získávání pozičních a časových dat. *GPS manager* zjednodušuje práci s využitou knihovnou, a omezuje její funkcionalitu na funkce, které jsou potřebné k funkci sledovacího zařízení. Modul *GPS manager* využívá jmenný prostor „GPS_MAN“.

5.3.4.1 *init*

Funkce *init* inicializuje fyzický GPS modul. Nastavuje rychlost přenosu dat mezi vývojovou deskou a modulem na 9600 baudů. Dále také nastavuje obnovovací frekvenci GPS modulu na 1 Hz.

5.3.4.2 *update*

Funkce *update* slouží ke čtení a ukládání dat přijatých fyzickým GPS modulem. Funkce zjišťuje, zda GPS modul přijal nová data. Pokud ano, pokouší se data validovat. Pokud jsou data validní, funkce vrátí hodnotu „true“, tím dá najevo, že jsou dostupná nová data. Jinak vrátí hodnotu „false“, což znamená, že nová data dostupná nejsou.

5.3.4.3 *fix*

Funkce *fix* slouží k předání stavu o tom, zda fyzický GPS modul aktuálně vypočítává svoji polohu. Pokud GPS modul nekomunikuje s dostatečným množstvím satelitů, nedokáže zjistit svoji polohu. V tomto případě funkce vrátí „false“. V případě, že satelit polohu zjistit dokáže, funkce vrátí „true“. Tato funkce pouze využívá knihovní funkci.

5.3.4.4 *getSats*

Funkce *getSats* slouží k získání počtu satelitů, se kterými aktuálně komunikuje fyzický GPS modul. Tato funkce pouze využívá funkci z použité knihovny.

5.3.4.5 *getLat*

Funkce *getLat* získává z ovládací knihovny fyzického GPS modulu zeměpisnou šířku. Získaná hodnota je uvedena ve stupních a minutách. Aby byla zeměpisná šířka ve formátu, který je využitý v datovém formátu GPX, musí být stupně a minuty odděleny. Minuty se poté vydělí číslem 60, a jsou přičteny ke stupňům. V případě, že se zařízení nachází na jižní polokouli je výsledná hodnota vynásobena číslem -1. Výsledná hodnota je funkcí vrácena.

5.3.4.6 *getLon*

Funkce *getLon* funguje podobně, jako funkce *getLan*. Rozdíl je, že pracuje se zeměpisnou délkou. Násobení hodnotou -1 je provedeno v případě, že se zařízení nachází na západní polokouli.

5.3.4.7 *getAlt*

Funkce *getAlt* využívá knihovní funkce k vrácení nadmořské výšky zařízení.

5.3.4.8 *getHour*

Funkce *getHour* využívá knihovní funkce k vrácení počtu hodin v současném dnu v časové zóně UTC.

5.3.4.9 *getMinute*

Funkce *getMinute* využívá knihovní funkce k vrácení počtu minut v současné hodině v časové zóně UTC.

5.3.4.10 *getSecond*

Funkce *getSecond* využívá knihovní funkce k vrácení počtu sekund v současné minutě v časové zóně UTC.

5.3.4.11 *getYear*

Funkce *getYear* využívá knihovní funkce k vrácení současného roku v časové zóně UTC. Rok je uveden pouze dvěma číslicemi.

5.3.4.12 *getMonth*

Funkce *getYear* využívá knihovní funkce k vrácení současného měsíce v časové zóně UTC.

5.3.4.13 *getDay*

Funkce *getYear* využívá knihovní funkce k vrácení současného dne v časové zóně UTC.

5.3.5 SD manager

Softwarový modul *SD manager* slouží k práci s fyzickým modulem čtečky SD karet. Obsahuje funkce určené k vytváření a úpravě souborů. Pro práci s hardwarovým modulem je využívána knihovna *SdFat*, která je určena pro práci s SD čtečkami. *SD manager* prací s knihovnou zjednodušuje, a omezuje její funkcionalitu pouze na funkce, které jsou potřebné pro tento projekt. Modul využívá jmenný prostor „SD_MAN“.

5.3.5.1 *init*

Funkce *init* inicializuje knihovní funkcí fyzický modul čtečky paměťových karet.

5.3.5.2 *openFileWrite*

Funkce *openFileWrite* slouží k otevírání souborů určených pro zápis dat. Funkce bere jako vstup řetězec určený k pojmenování souboru. Funkce otevře soubor s vybraným jménem. V případě, že soubor neexistuje, je vytvořen. V případě, kdy je soubor úspěšně otevřen, funkce vrací hodnotu „true“. Pokud došlo k chybě, je vrácena hodnota „false“.

5.3.5.3 *fileWrite*

Funkce *fileWrite* pomocí knihovní funkce zapisuje textová data do aktuálně otevřeného souboru. Jako vstup bere textový řetězec, který bude zapsán do souboru. Data jsou přidána do souboru na současnou pozici. V případě, že za současnou pozici již existuje text, je přepsán.

5.3.5.4 *fileFlush*

Funkce *fileFlush* pomocí knihovní funkce zajišťuje, že jsou data zapsána do aktuálně otevřeného souboru. Ideálně by se měla volat po úpravách souboru.

5.3.5.5 *fileClose*

Funkce *fileClose* pomocí knihovní funkce zavírá aktuálně otevřený soubor. Po zavření do souboru nelze psát.

5.3.5.6 *fileExists*

Funkce *fileExists* pomocí knihovní funkce kontroluje, zda vybraný soubor existuje. Funkce přijímá textový řetězec se jménem souboru. Pokud soubor existuje, funkce vrací hodnotu „true“. Pokud soubor neexistuje, vrací „false“.

5.3.5.7 *seek*

Funkce *seek* pomocí knihovní funkce mění současnou pozici v aktuálně otevřeném souboru. Funkce jako vstup bere číslo bajtu, který bude sloužit jako nová pozice. Bajty jsou číslovány od začátku po konec souboru, pokud je zadán bajt 0, nová pozice bude na začátku souboru. Po přesunutí na novou pozici jsou nová data zapsána od této pozice. Stará data na následujících pozicích jsou novými daty přepsána.

5.3.5.8 *seekBack*

funkce *seekBack* je inverze funkce *seek*. Využívá knihovní funkci k přesunu na žádanou pozici v aktuálně otevřeném souboru. Jako vstup bere číslo bajtu, který bude sloužit jako nová

pozice. Na rozdíl od funkce *seek* jsou bajty číslovány od konce souboru. Bajt 0 je v tomto případě poslední bajt souboru.

5.3.6 position logger

Softwarový modul *position logger* je určený k ukládání pozičních dat pomocí modulu *SD manager*. Modul pomocí *SD manageru* ukládá poziční data získané z hlavního modulu *main* do souboru na paměťové kartě. Data projdou přes modul *GPX generator*, který je převede do správného formátu. Softwarový modul používá jmenný prostor *POS_LOG*.

5.3.6.1 init

Funkce *init* zavolá inicializační funkce softwarového modulu *SD manager*. Slouží k inicializaci fyzického modulu čtečky *SD karet*.

5.3.6.2 startLogging

Funkce *startLogging* je určena k přípravě a započetí ukládání pozičních dat v nové sledovací relaci. Funkce nejdříve vytvoří zásobník s textovým řetězcem „0.gpx“, který slouží jako výchozí jméno nového trasového souboru. Funkce poté jde do smyčky. V této smyčce testuje, zda jméno v zásobníku již na paměťové kartě existuje. Pokud existuje, zvyšuje číslo v řetězci. Číslo je zvyšováno, dokud není nalezeno originální jméno. V případě že číslo dosáhne šestnáctibitového limitu funkce vrací hodnotu „false“. Po nalezení originálního jména se funkce snaží soubor otevřít k zápisu. V případě, že soubor nebyl úspěšně otevřený funkce vrací hodnotu „false“. Funkce následně vytvoří dva nové textové zásobníky. První zásobník o velikosti 50 znaků slouží jako vstup do funkce softwarového modulu *GPX generator*. Druhý zásobník o velikosti 250 znaků slouží k uložení výstupu ze stejné funkce. Po vytvoření zásobníků je zavolána zmíněná funkce softwarového modulu *GPX generator*. Tato funkce vygeneruje a vrátí kostru souboru formátu *GPX*. Funkce také vrátí počet bajtů od konce souboru určený k posunu na volné místo, kde budou ukládány trasové body. V případě chyby generace je vrácena hodnota 0, v tomto případě vrátí funkce *startLogging* hodnotu „false“. Pokud vše proběhne úspěšně, je nový obsah zapsán do souboru, funkce vrací hodnotu „true“.

5.3.6.3 addTrackpoint

Funkce *addTrackpoint* slouží k přidání nového trasového bodu ze současné sledovací relace do aktuálně otevřeného souboru na paměťové kartě. Funkce bere velké množství vstupních

proměnných, které slouží jako hodnoty uložené v trasovém souboru. Tyto proměnné obsahují informace o aktuální zeměpisné šířce a délce, aktuální výšce nad mořem v metrech a o aktuálním datu a času v časové zóně UTC. Funkce nejdříve vytvoří textový zásobník o velikosti 150 znaků. Poté volá funkci ze softwarového modulu *GPX generator*, která slouží k vygenerování textové reprezentace trasového bodu. Trasový bod je uložen do vytvořeného zásobníku. V případě chyby v generaci vrací funkce z generátoru hodnotu „false“. V tomto případě funkce *addTrackpoint* také vrací hodnotu „false“. V případě úspěšné generace jsou data připravena k zápisu do souboru. Nejdříve se pozice v souboru posouvá dozadu, na místo vyhrazené pro zápis nového trasového bodu. Data od tohoto bodu jsou přepsána novým trasovým bodem. Změny jsou nakonec uloženy do souboru, funkce vrací hodnotu „true“.

5.3.6.4 *stopLogging*

Funkce *stopLogging* slouží k ukončení ukládání pozičních dat z ukončené sledovací relace. Funkce zavře aktuálně otevřený soubor a nastaví hodnotu určenou k posunu v souboru na nulu.

5.3.7 **GPX generator**

Softwarový modul *GPX generator* přijímá poziční data, která poté převádí na části, ze kterých se sestavuje výsledný trasový soubor ve formátu GPX. Softwarový modul používá jmenný prostor `GPX_GEN`.

5.3.7.1 *generateSkeleton*

Funkce *generateSkeleton* slouží k vygenerování základní „kostry“ souboru typu GPX. Funkce přijímá řetězce se jménem autora trasy a se jménem trasy samotné. Přijímá také textový zásobník a informace o jeho velikosti. Tento zásobník slouží k uložení výsledné vygenerované kostry. Funkce nejdříve kontroluje, zda je zásobník dostatečně velký. V případě, že zásobník nesplňuje velikostní požadavek je vrácena hodnota 0, tímto se předejde zápisu do paměti mimo zásobník. Při dostatečné velikosti se pomocí funkce k vytvoření formátovaného řetězce do zásobníku uloží kostra souboru typu GPX, která obsahuje zadané informace o jméně a autorovi trasy. Funkce vrací počet bajtů potřebných k posunu od konce souboru k volnému místu vyhrazenému pro ukládání trasových bodů.

```

<?xml version="1.0" encoding="utf-8"?>
<gpx xmlns="http://www.topografix.com/GPX/1/1" version="1.1" creator="Martin Machala - https://github.com/m-machala">
  <trk>
    <name>GPS Tracker - track 32</name>
    <trkseg>
      </trkseg>
    </trk>
  </gpx>

```

Obrázek 16: Vygenerovaná kostra souboru

5.3.7.2 *generateTrackpoint*

Funkce *generateTrackpoint* slouží k vygenerování trasového bodu souboru typu GPX. Funkce bere velké množství vstupů určených k vytvoření trasového bodu. Tyto proměnné obsahují informace o aktuální zeměpisné šířce a délce, aktuální výšce nad mořem v metrech a současné datum a čas v časové zóně UTC. Mezi vstupy také patří textový zásobník a informace o velikosti tohoto zásobníku. Funkce nejdříve zkontroluje, zda je zásobník dostatečně veliký. V případě, že není, je vrácena hodnota „false“. Pokud je zásobník dostatečně veliký, funkce pokračuje. Pro korektní zápis čísel s desetinnou tečkou je potřeba převést tato čísla pomocí funkce *dtostrf* na textové řetězce. Tyto řetězce jsou uloženy v nově vytvořených textových zásobnících. Funkce pomocí funkce *k* vytvoří formátovaného řetězce ukládá trasový bod s požadovanými informacemi do vstupního zásobníku. Funkce nakonec vrací hodnotu „true“.

```

<trkpt lat="49.219231" lon="17.660036">
  <ele>21.000</ele>
  <time>2023-03-02T10:30:07Z</time>
</trkpt>

```

Obrázek 17: Vygenerovaný trasový bod

5.3.8 *main*

Hlavní softwarový modul *main* slouží k ovládní celého zařízení pro záznam trasy uživatele. Ke své funkci využívá softwarové moduly *GPS manager*, *position logger* a *GUI manager*. Tyto moduly jsou napojené na ostatní softwarové moduly vytvořené pro sledovací zařízení. Modul *main* se dělí na dvě hlavní funkce. Dále využívá dvě vedlejší pomocné funkce.

5.3.8.1 *systemError*

Pomocná funkce *systemError* je volána v případě, že v systému došlo k chybě, kterou nelze softwarově napravit. Funkce v nekonečné smyčce periodicky bliká vnitřní LED diodou vývojové desky.

5.3.8.2 *readButton*

Pomocná funkce *readButton* slouží ke čtení stavu tlačítka. Jako vstupní hodnotu funkce bere číslo pinu tlačítka. Funkce nejdříve zkouší, zda je tlačítko zmáčknuté. V případě, že ano, funkce počká deset milisekund a tlačítko znovu testuje. Pokud je tlačítko stále zmáčknuté, vrací hodnotu „true“. V ostatních případech vrací hodnotu „false“. Opakované čtení stavu tlačítka chrání před nežádoucími falešnými pozitivy při čtení tlačítka.

5.3.8.3 *setup*

Funkce *setup* je volána pouze jedenkrát při zapnutí zařízení. Slouží převážně k inicializaci a nastavení modulů. Zobrazuje také úvodní obrazovku zařízení.

Funkce se nejdříve pokusí inicializovat softwarový modul *GUI manager*, který slouží k jednoduché práci s uživatelským prostředím. V případě, že modul není úspěšně inicializován, je zavolána funkce *systemError*.

Po úspěšné inicializaci je na displeji na pět sekund zobrazena úvodní obrazovka zařízení, po které je vykresleno základní rozložení obrazovky.

Po vykreslení uživatelského prostředí jsou inicializovány moduly *GPS manager* a *position logger*. V případě chyby inicializace modulu se na displeji vykreslí varovná zpráva s informacemi pro uživatele. Následně je spuštěna funkce *systemError*.

Po úspěšných inicializacích jsou nastaveny dva vstupní piny pro tlačítka určené k ovládání zařízení.

Nakonec je vykreslena obrazovka s informacemi o úspěšné inicializaci a s instrukcemi k započatí sledovací relace.

5.3.8.4 *loop*

Funkce *loop* je nekonečná smyčka volána po úspěšném dokončení funkce *setup*. Tato smyčka by za normálních okolností neměla nikdy být ukončena. Slouží jako hlavní logika programu.

Funkce se nejdříve snaží zjistit, zda fyzický GPS modul obsahuje nová data k přečtení. Pokud ano, tak funkce aktualizuje zobrazené informace o současném datu, čase a počtu satelitů.

Následná část funkce používá návrh ve stylu konečného automatu. Funkce se pomocí switche dělí do několika větví. Vstup switche je současný stav programu. Existuje pět různých

stavů, ve kterých program může být. Tato část kódu je volána jenom přibližně jedenkrát za 100 milisekund.

Funkcionalita možných stavů je popsána níže.

5.3.8.4.1 STARTING

Stav STARTING slouží k započatí nové sledovací relace.

Program nejdříve kontroluje, zda bylo zmáčknuté tlačítko 1. Pokud tlačítko nebylo zmáčknuté, nic se nestane, a program pokračuje za konečným automatem.

Pokud tlačítko zmáčknuté bylo, program zkouší, zda fyzický GPS modul již určil polohu zařízení.

Pokud poloha určena nebyla, je na displeji vypsána informační zpráva o tom, že zařízení nemá GPS signál. Zařízení se přesune do stavu NOFIX.

Pokud poloha určena byla, program se pokusí začít zapisovat trasová data do nového souboru.

V případě, že nastala při generaci souboru chyba, je vypsána chybová hláška, a je zavolána funkce *systemError*.

Při úspěšném vytvoření souboru jsou resetovány proměnné určené k ukládání statistik o sledovací relaci. Také jsou nastaveny proměnné, které ukládají pozici a čas předchozího zavolání funkce.

Program se nakonec přesune do stavu TRACKING.

5.3.8.4.2 NOFIX

Stav NOFIX čeká 5 sekund, což uživateli umožňuje přečíst informační zprávu zobrazenou v předchozím stavu. Čekání neomezí ostatní funkce zařízení mimo konečný automat, informace o datu, čase a počtu satelitů jsou tak stále aktualizované.

Po skončení čekací doby je znovu zobrazena obrazovka s informacemi o inicializaci a započatí nové relace. Program je přesunut do stavu STARTING.

5.3.8.4.3 TRACKING

Stav TRACKING slouží k ukládání pozičních dat na paměťovou kartu.

Funkce nejdříve zjistí, jaký je současný celkový počet minut ve dni. Hodnotu pak porovnává s předchozím uloženým počtem minut.

Pokud je současný počet minut vyšší než ten předchozí, je jejich rozdíl přičtený k počítadlu minut od začátku sledování. V případě, že nastal nový den, je rozdíl současných minut a předchozích minut negativní. V tomto případě je k současnému počtu minut přičteno 1440 minut, což odpovídá jednomu dnu. Je provedena stejná operace.

Následně program zkouší, zda jsou dostupná nová poziční data. Zároveň také zkouší, zda fyzický GPS modul vypočítal v získaných datech pozici zařízení. Pokud jsou obě podmínky splněny, je do souboru přidán nový trasový bod. Sledovací obrazovka je aktualizována s novými pozičními daty.

Program také zkouší, zda bylo zmáčknuto tlačítko 1. Pokud ano, aktualizuje sledovací obrazovku s instrukcemi k přerušení čekání a přesune program do stavu WAITING.

5.3.8.4.4 WAITING

Stav WAITING je stav pozastavení sledovací relace.

Program nejdříve kontroluje, zda bylo zmáčknuto tlačítko 1. V případě, že bylo, je aktualizována hodnota předchozího počtu minut. Následně je automat přesunut do stavu TRACKING.

Program dále kontroluje, zda bylo zmáčknuto tlačítko 2. V tomto případě zobrazí konečnou obrazovku, ukončí sledovací relaci a přesune se do stavu ENDING.

5.3.8.4.5 ENDING

Stav ENDING čeká, než uživatel zmáčkne tlačítko 1. V tomto případě je program přesunut do stavu STARTING.

5.4 Hodnocení finální realizace

Finální realizace zařízení pro záznam trasy uživatele splňuje všechny požadavky, které byly v této práci stanoveny. Je složena z komponentů, které lze jednoduše získat, a které jsou cenově dostupné. Software sledovacího zařízení je jednoduchý k použití, a zobrazuje informace o případných chybách, které mohou nastat. Hardwarový návrh i software tohoto zařízení je volně dostupný na platformě GitHub: <https://github.com/m-machala/Arduino-GPS->

Tracker. Software je dostupný pod permissivní licenci MIT. Kdokoli si tak může vytvořit vlastní kopii zařízení, nebo vytvořit vlastní vývojovou větev, kde software sám upraví.

Zařízení na paměťové kartě úspěšně generuje soubory formátu GPX, které jsou kompatibilním softwarem čitelné. Za normálních podmínek (tj., zařízení „vidí“ nebe) je přesnost měření v okolí 2-3 metrů od reálné polohy, a nová poloha je získávána přibližně jednou za sekundu. Sledovací zařízení tak překonává množství dražších chytrých zařízení.

Nedostatkem zařízení je absence uživatelsky dostupných možností pro nastavení obslužného softwaru. Současná realizace zařízení také využívá samostatné moduly propojené jumper kabely. Ideálně by zařízení mohlo fungovat jako jeden plošný spoj, což by výrazně zvýšilo odolnost zařízení. Pravděpodobně by takovéto rozhodnutí zvýšilo konečnou cenu zařízení.

6 PROGRAM PRO ANALÝZU TRASY UŽIVATELE

Program pro analýzu trasy uživatele slouží ke zpracování a výpisu informací o trasách vygenerovaných zařízením pro záznam trasy uživatele. Program vypisuje statistiky a generuje grafy z poskytnutých dat.

Tento program byl vytvořený v programovacím jazyku Python v programovacím prostředí PyCharm. V programu byly využity tři externí knihovny.

Knihovna PyQt5 slouží k jednoduchému vytváření uživatelského prostředí napříč operačními systémy.

Knihovna matplotlib slouží ke generování grafů.

Knihovna gpxpy slouží ke zjednodušení práce s datovým formátem GPX.

Na rozdíl od softwarového vybavení zařízení pro záznam trasy není tento program dělen na moduly, existuje pouze hlavní soubor, který obsahuje několik funkcí.

6.1 main

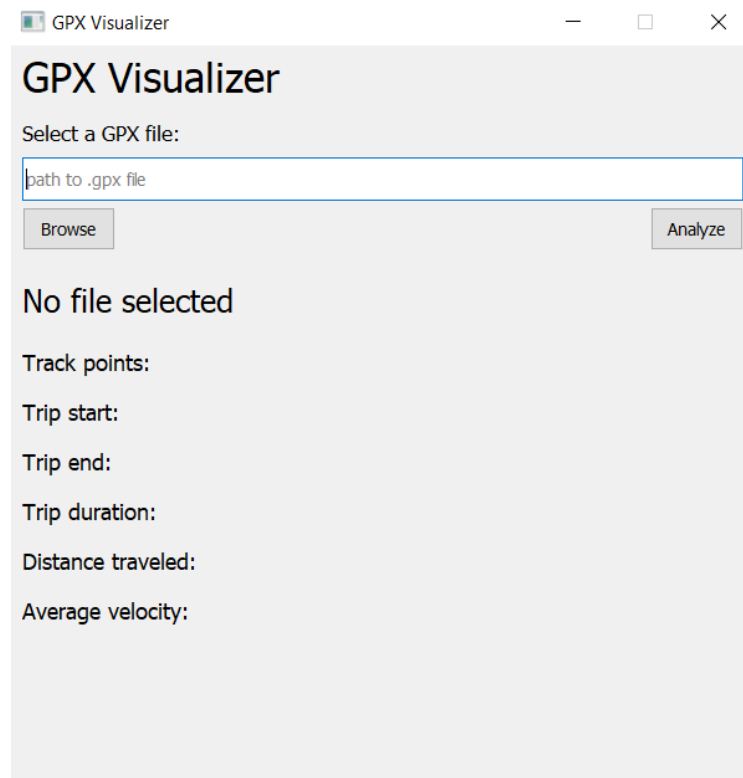
Funkce *main* slouží k inicializaci uživatelského prostředí pomocí funkcí knihovny PyQt5. Nejdříve je vytvořena instance aplikace, poté instance třídy *Window*. Uživatelské prostředí je následně zobrazeno.

6.2 Window

Třída *Window* obsahuje informace o rozložení uživatelského prostředí a funkce, které slouží k práci s daty v uživatelském prostředí.

6.2.1 __init__

Funkce *__init__* je volána jedenkrát při vytvoření instance třídy. V této funkci jsou definovány všechny vlastnosti a elementy uživatelského prostředí.



Obrázek 18: Uživatelské prostředí po inicializaci instance třídy

Nejdříve je nastaveno jméno okna, jeho výška a šířka.

Následně je vytvořen štítek se jménem aplikace, a je upraven formát textu štítku.

Poté je vytvořena oblast pro vkládání cesty k souboru ve formátu GPX. Tato oblast obsahuje štítek s instrukcemi, textové pole, kde se vkládá cesta a dvě tlačítka, která slouží k ovládání programu.

Tlačítko *Browse* pomocí funkce *fileSelect* otevírá systémový prohlížeč souborů pro jednoduchý výběr souboru.

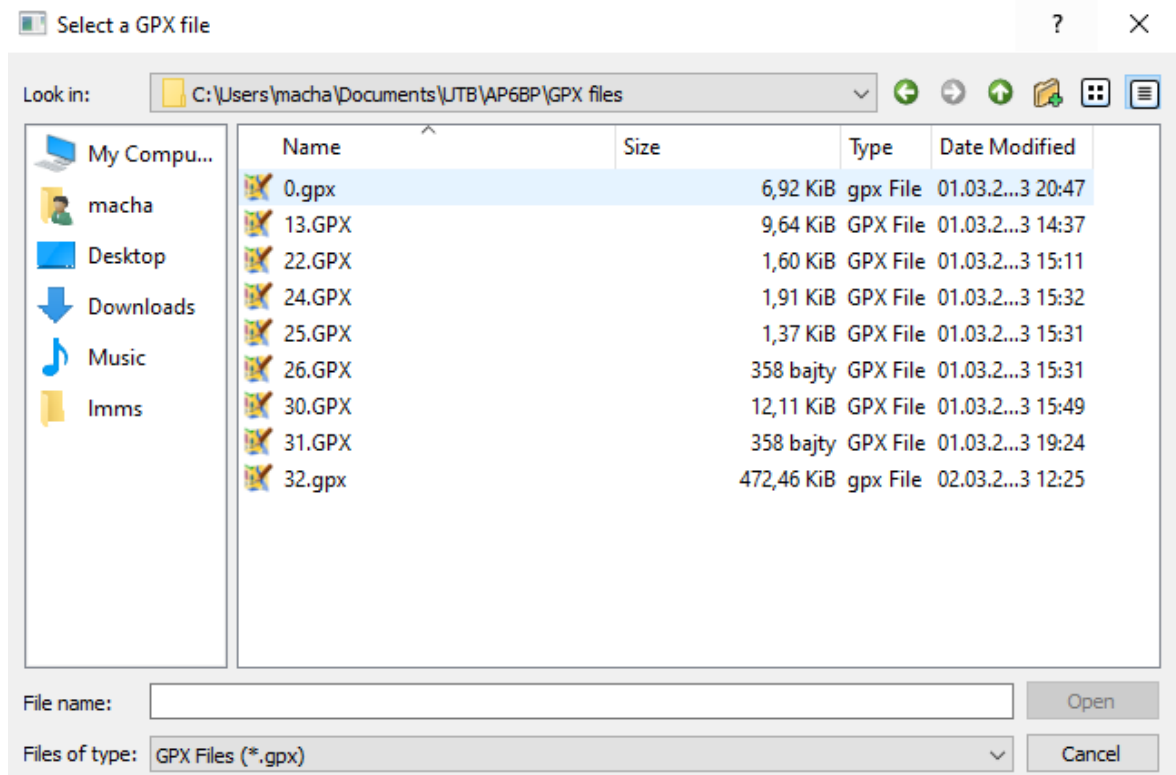
Tlačítko *Analyze* pomocí funkce *analyzeData* spouští analýzu vybraného souboru, po které jsou vyplněny pole určené k výpisu informací. Také se zobrazí grafy s vizualizacemi dat.

Nakonec je vytvořena sekce určená k výpisu informací získaných z trasového souboru. Tato sekce obsahuje štítek, do kterého se vypisuje jméno aktuálního souboru, popřípadě chybová hláška, pokud nastane chyba.

Sekce obsahuje informace o počtu trasových bodů v souboru, datum a čas začátku a konce sledovací relace, časovou délku relace, počet odcestovaných kilometrů ve sledovací relaci a průměrnou rychlost v relaci.

6.2.2 fileSelect

Funkce *fileSelect* slouží k otevírání systémového dialogového okna určeného pro výběr souborů. Dialogové okno je nastaveno tak, aby bralo pouze soubory ve formátu GPX. Předchází tak možným chybám, které mohou nastat vložení souboru s nesprávným formátem. Po výběru souboru je cesta k tomuto souboru uložena do textového pole určeného pro cestu k souboru formátu GPX.



Obrázek 19: Dialogové okno pro zvolení souboru ve formátu GPX

6.2.3 analyzeData

Funkce *analyzeData* slouží k analýze a zpracování pozičních dat získaných ze souboru ve formátu GPX.

Funkce se nejdříve snaží otevřít soubor, jehož cesta byla zadána do textového pole pro zadání cesty k souboru. V případě, že tato cesta nevede k žádnému souboru je vypsána chybová hláška „File not found!“. V případě, že byly zobrazena data o jiném souboru, jsou tato data smazána.

V případě úspěšného otevření souboru je pomocí knihovny gpxpy analyzován. Pokud soubor není možné analyzovat, například z důvodu chyby v souboru, je vypsána chyba „Invalid file!“. Pokud byly zobrazena data o jiném souboru, jsou tato data smazána.

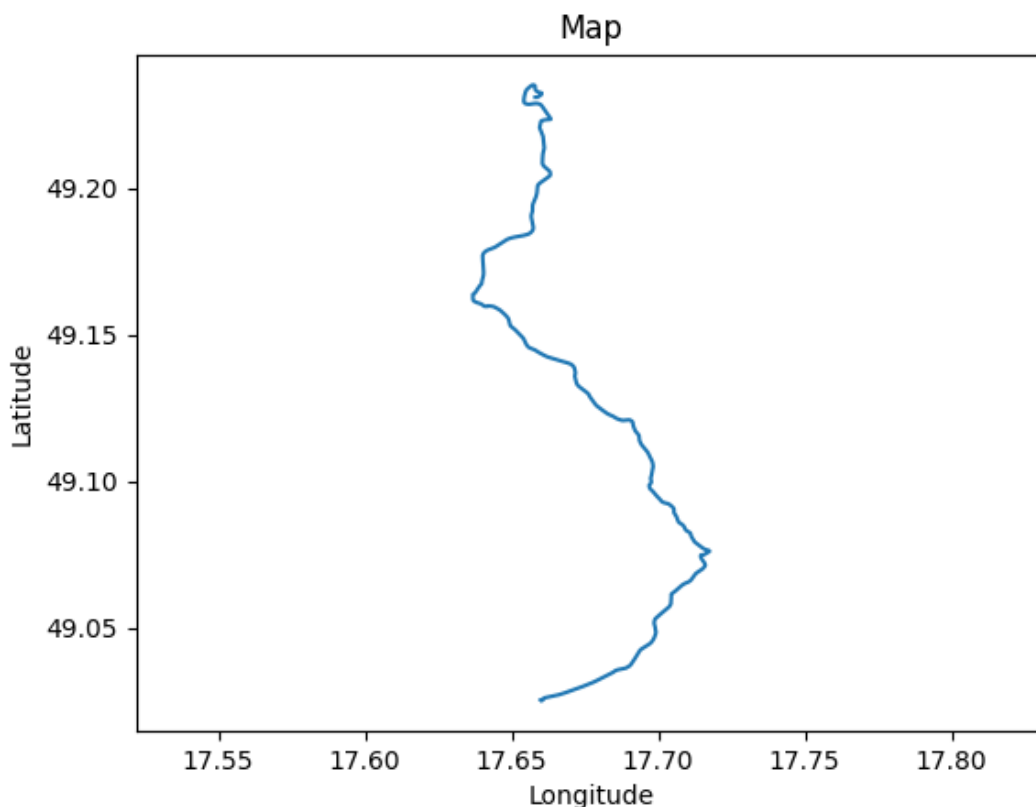
Po úspěšné analýze souboru se zobrazí jméno uložené v souboru.

Funkce si vytvoří množství proměnných určených pro data k vytváření grafů z dat souboru.

Následně funkce prochází všemi body trasy uložené v souboru. Zeměpisná šířka a délka každého bodu je ukládána do listů. Pomocí funkce z knihovny gpxpy je zjištěna vzdálenost a časový rozdíl mezi bodem současným a předchozím, Tyto hodnoty jsou přičteny k proměnným, které obsahují celkovou vzdálenost. Pokud mezi dvěma body nedošlo ke změně času, je jejich vzdálenost uložena, a přičtena k následujícímu bodu. Aktuální sečtená vzdálenost je postupně ukládána do listu, zároveň s časem, kdy bylo této vzdálenosti dosaženo. Stejně tak se ukládá i rychlost zařízení a čas, kdy této rychlosti bylo dosaženo.

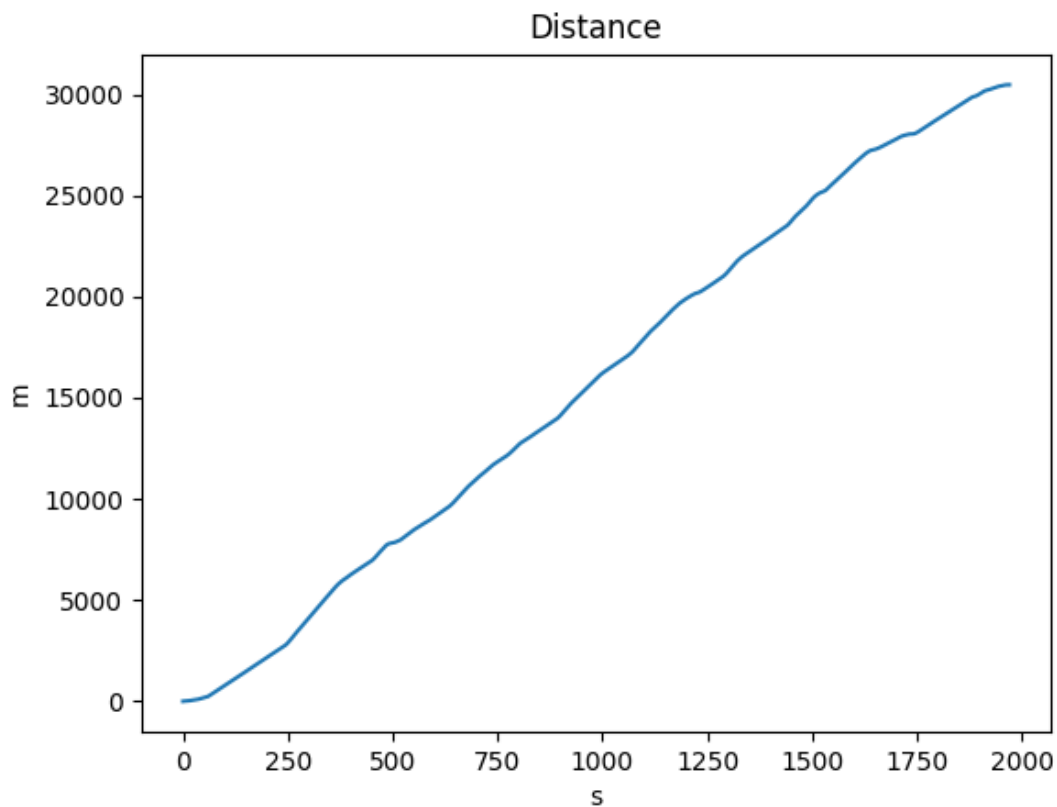
Po získání všech potřebných dat se pomocí knihovny matplotlib generují grafy s těmito daty.

Nejdříve je generován graf, který zobrazuje všechny souřadnice v trase. Souřadnice tvoří mapu cesty uživatele. Osy grafu obsahují zeměpisnou šířku a zeměpisnou délku.



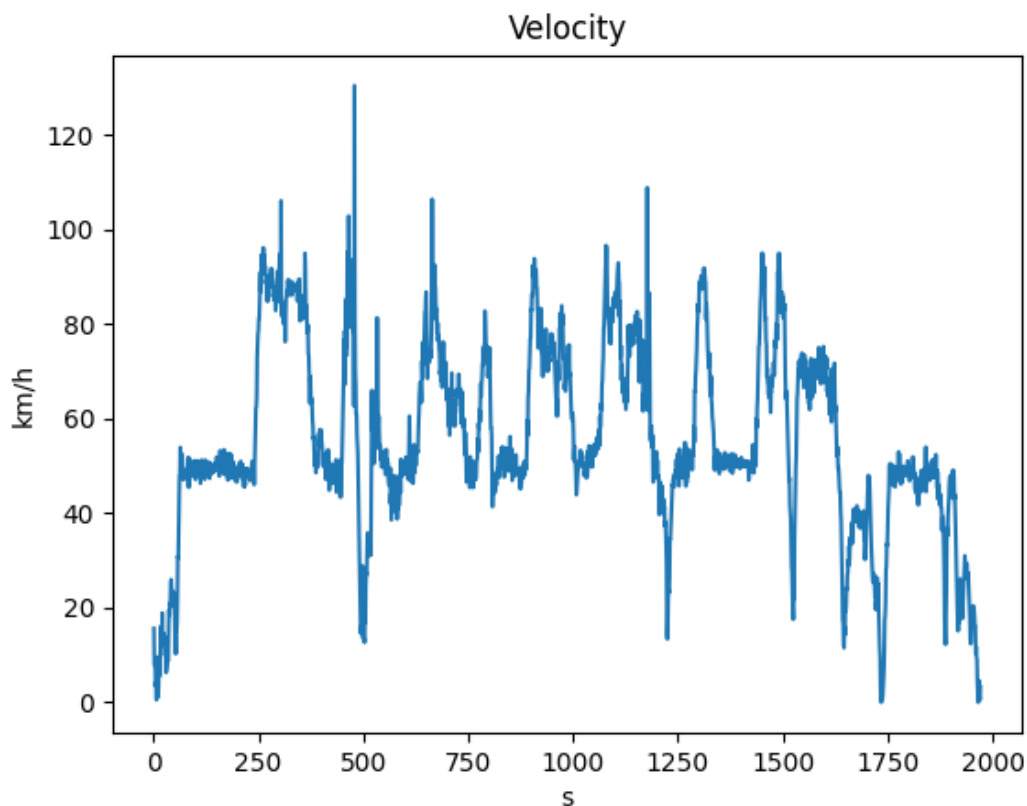
Obrázek 20: Graf zobrazující trasu cesty autem mezi městy Uherský Brod a Zlín, ygenerován programem pro analýzu trasy uživatele. Data získána ze zařízení pro záznam trasy uživatele

Následně je generován graf, který ukazuje, jaké vzdálenosti bylo v určitý čas dosaženo. Vzdálenost je uvedena v metrech, čas v sekundách.



Obrázek 21: Graf zobrazující ujetou vzdálenost trasy mezi městy Uherský Brod a Zlín, vygenerován programem pro analýzu trasy uživatele. Data získána ze zařízení pro záznam trasy uživatele.

Jako poslední je vygenerován graf, který ukazuje rychlost, které bylo v určitém čase dosaženo. Rychlost je uvedena v kilometrech za hodinu, čas je uveden v sekundách.

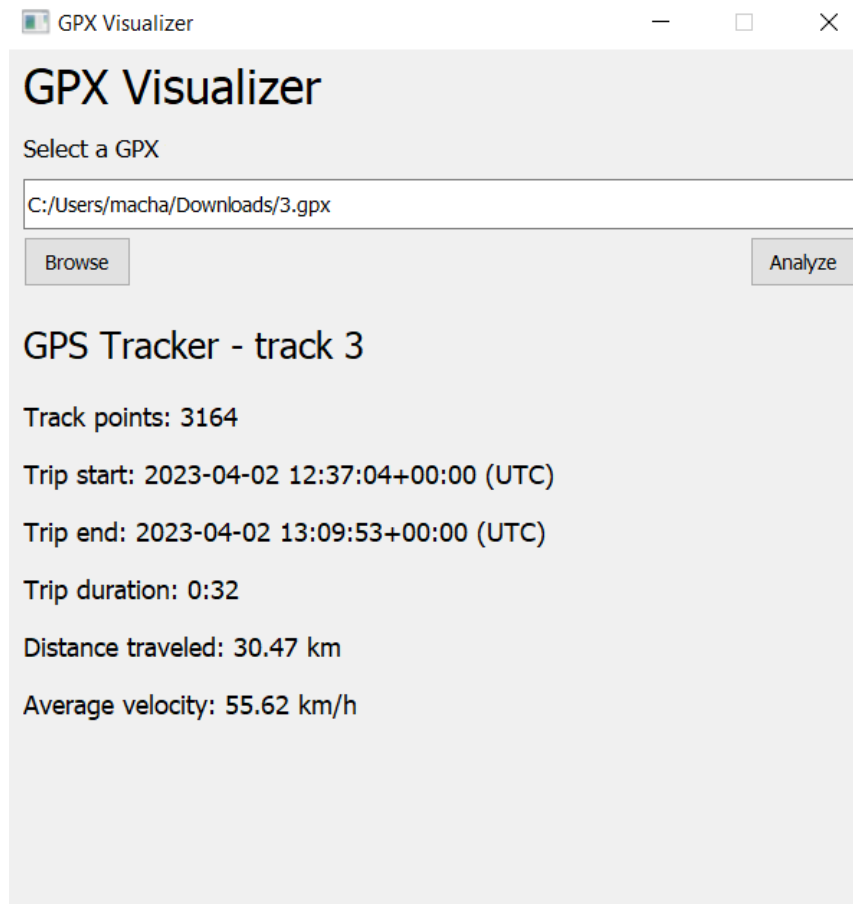


Obrázek 22: Graf zobrazující rychlost pohybu zařízení na trase mezi městy Uherský Brod a Zlín, vygenerován programem pro analýzu trasy uživatele. Data získána ze zařízení pro záznam trasy uživatele.

Všechny tyto grafy jsou zobrazeny ve svých vlastních oknech vygenerovaných knihovnou matplotlib. Okna obsahují ovládací prvky pro jednoduchou práci s grafy.

Po zobrazení grafů program generuje ostatní informace o trase. Pomocí velikosti listu se souřadnicemi zjistí počet trasových bodů. Sebráním času prvního a posledního bodu získá časové známky začátku a konce sledovací relace. Pomocí funkce knihovny gpxpy zjistí časový rozdíl mezi těmito body. Rozdíl je nakonec převeden na hodiny a minuty.

Všechny vygenerované informace jsou zobrazeny v hlavním okně uživatelského prostředí.



Obrázek 23: Uživatelské prostředí s vygenerovanými informacemi o cestě mezi městy Uherský Brod a Zlín. Data získána ze zařízení pro záznam trasy uživatele.

6.2.4 eraseData

Funkce *eraseData* slouží k vymazání dat vygenerovaných uživatelským prostředím. Všechny hodnoty jsou nahrazeny prázdnými textovými řetězci.

6.3 Hodnocení finální realizace

Finální realizace programu pro analýzu trasy uživatele splňuje stanovené požadavky této práce. Aplikace poskytuje multiplatformní řešení pro analýzu a vizualizaci dat získaných ze zařízení pro záznam trasy uživatele. Kód programu je veřejně dostupný na platformě GitHub: <https://github.com/m-machala/GPX-Visualizer>. Kód je dostupný pod permissivní licenci MIT. Kdokoli program může stáhnout, využít nebo vytvořit vlastní vývojovou větev, kde program sám upraví.

Aplikace úspěšně zpracovává a analyzuje poskytnuté soubory s trasovými daty. Vypisuje informace o trase v podobě datových bodů a grafů.

Nedostatek této aplikace je neúplná kompatibilita se soubory s více trasami. Program pracuje pouze s první trasou uvedenou v trasovém souboru. Toto by nemělo způsobovat problémy s daty získanými ze zařízení pro záznam trasy uživatele, ve kterém jsou všechna data v jednom souboru uložena pouze jako jedna trasa.

ZÁVĚR

Cílem této práce bylo navrhnout, sestavit a naprogramovat zařízení pro záznam trasy uživatele a navrhnout a naprogramovat program pro analýzu trasy uživatele.

Teoretická část práce se zaměřuje především na vypracování literární rešerše. Cílem této rešerše je popsat existující řešení problematiky této práce. První část rešerše mluví o různých existujících řešeních a realizacích zařízení pro záznam trasy uživatele. Různá řešení jsou porovnávána jak výkonově, tak cenově. Data pro tuto část práce byla získána ze zdrojů, které poskytují informace ohledně ceny a výkonu různých sledovacích zařízení. Druhá část ukazuje různá řešení a realizace softwaru, který lze využít pro analýzu trasových dat poskytnutých uživatelem. Vybraná řešení pro tuto část jsou služby Mapy.cz, Google Earth a GPS Visualizer. Informace o těchto službách byly získány použitím služeb samotných a také z dokumentace, která je pro tyto služby dostupná.

Přínosem této rešerše bylo získání informací ohledně funkcionality, výhod i nevýhod existujících řešení, jak pro systémy pro záznam trasy uživatele, tak i pro systémy pro analýzu trasy uživatele.

Teoretická část také popisuje technologii GPS, princip její funkce a přesnost systému. V této části jsou také popsány jiné poziční systémy a datový formát GPX, který je v této práci využíván k ukládání trasových dat.

Jako poslední popisuje teoretická část vývojovou platformu Arduino. Zabývá se jejím využitím, hardwarem a také o nástrojem PlatformIO, který je v této práci využíván pro vývoj softwaru pro platformu Arduino.

První polovina praktické části popisuje návrh, sestavu a softwarovou implementaci zařízení pro záznam trasy uživatele. Cílem tohoto zařízení je zaznamenávání trasy uživatele za pomoci GPS přijímače a ukládání těchto dat ve formátu GPX na paměťovou kartu. Zařízení by také mělo obsahovat uživatelské rozhraní které je uživatelsky přívětivé. Požadavky na zařízení jsou v praktické části definovány a vysvětleny. Následně jsou vybrány komponenty, ze kterých je zařízení složeno.

Konečná sestava zařízení se skládá z klonu vývojové desky Arduino Mega 2560, GPS modulu NEO-7M, jednobarevného OLED displeje, čtečky microSD karet a dvou tlačítek. Po návrhu sestavy je popsáno obslužné softwarové vybavení, které bylo pro toto zařízení vyvinuté. Práce popisuje vytvořené softwarové moduly, jejich funkcionalitu a funkce. Software

byl vyvinutý v programovacím jazyce C++ za pomoci vývojového prostředí Visual Studio Code, do kterého bylo přidáno rozšíření PlatformIO, které zjednodušuje vývoj softwaru pro embedded zařízení. Při vývoji softwaru byly použité knihovny, které slouží k práci s využitými komponenty. Mezi tyto knihovny patří knihovna Adafruit_SH110X, která slouží k ovládní OLED displejů řady SH110X, knihovna Adafruit_GPS, která slouží k ovládní GPS modulů a knihovna SdFat, která slouží k ovládní čteček paměťových karet. Samotný software se skládá z osmi softwarových modulů, které jsou mezi sebou propojené.

Finální realizace zařízení pro záznam trasy uživatele splňuje stanovené požadavky. Zařízení se skládá ze snadno přístupných a cenově dostupných komponentů. Je schopné generovat soubory typu GPX, které obsahují trasová data nasbíraná GPS přijímačem. Přesnost získaných dat ze sledovacího zařízení je v okolí 2-3 metrů od reálné pozice sledovacího zařízení. Tato data jsou ukládána na paměťovou kartu. Uživatelské rozhraní zařízení je uživatelsky přívětivé a jednoduché k používání. Hardwarový návrh i software je veřejně dostupný pod permissivní licenci MIT.

Realizace nabízí možnosti pro budoucí rozšíření a úpravy. Uživatelské prostředí by mohlo nabízet možnosti pro nastavení funkcionality zařízení. Současné provedení propojení vývojové desky a hardwarových komponentů není příliš stabilní, ideálně by zařízení mohlo fungovat jako plošný spoj. Tato změna by ale pravděpodobně zvýšila výrobní cenu zařízení.

Druhá polovina praktické části popisuje vývoj a funkcionality programu pro analýzu trasy uživatele. Cílem tohoto programu je práce s trasovými daty vygenerovanými zařízením pro záznam trasy uživatele ve formátu GPX. Tato data mají být přijata a následně zpracována, výsledky mají být vypsány a vizualizovány pomocí grafů. Program byl napsán v programovacím jazyce Python za pomoci programovacího prostředí PyCharm. Při vývoji aplikace byly využity knihovny. Tyto knihovny jsou PyQt5, knihovna pro jednoduché vytváření uživatelského prostředí, Matplotlib, knihovna pro generování grafů a gpxpy, knihovna pro práci se soubory typu GPX.

Finální realizace softwaru pro analýzu trasy uživatele také splňuje stanovené požadavky. Program přijímá trasové soubory formátu GPX, které analyzuje, vypisuje a vizualizuje pomocí datových bodů a grafů. Software je také uveden pod permissivní licenci MIT.

Nedostatkem programu je snížená kompatibilita s GPX soubory obsahující záznamy o více trasách. Tento nedostatek ale nepůsobí žádné problémy při analýze dat vygenerovaných sledovacími zařízeními.

SEZNAM POUŽITÉ LITERATURY

- [1] GPS Overview. *GPS.gov* [online]. Washington, D.C.: National Coordination Office for Space-Based Positioning, Navigation, and Timing [cit. 2023-03-11]. Dostupné z: <https://www.gps.gov/systems/gps/>
- [2] Control Segment. *GPS.gov* [online]. Washington, D.C.: National Coordination Office for Space-Based Positioning, Navigation, and Timing [cit. 2023-03-11]. Dostupné z: <https://www.gps.gov/systems/gps/space/>
- [3] GPS Accuracy. *GPS.gov* [online]. Washington, D.C.: National Coordination Office for Space-Based Positioning, Navigation, and Timing [cit. 2023-03-12]. Dostupné z: <https://www.gps.gov/systems/gps/performance/accuracy/>
- [4] FOSTER, Dan. GPX Explained. *TopoGrafix* [online]. Stow (Massachusetts): Foster, c1998–2023 [cit. 2023-03-15]. Dostupné z: https://www.topografix.com/gpx_for_users.asp
- [5] FOSTER, Dan. GPX: the GPS Exchange Format. *TopoGrafix* [online]. Stow (Massachusetts): Foster, c1998–2023 [cit. 2023-03-15]. Dostupné z: <https://www.topografix.com/gpx.asp>
- [6] Other Global Navigation Satellite Systems (GNSS). *GPS.gov* [online]. Washington, D.C.: National Coordination Office for Space-Based Positioning, Navigation, and Timing [cit. 2023-03-17]. Dostupné z: <https://www.gps.gov/systems/gnss/>
- [7] GPS on smartphones: Testing the accuracy of location positioning. *DXOMARK* [online]. Boulogne-Billancourt: DXOMARK, c2008-2023, February 10, 2023 [cit. 2023-03-18]. Dostupné z: <https://www.dxomark.com/gps-on-your-smartphone-why-youre-not-always-there-when-it-says-youre-there/>
- [8] ELLINGS, John a Jenny ABEGG. Best Handheld GPS of 2023. *Switchback Travel* [online]. Switchback Travel, c2023 [cit. 2023-03-22]. Dostupné z: <https://www.switchbacktravel.com/best-handheld-gps>
- [9] Best GIS Data Collection Tools: GPS Devices & Digitizers. *Unearth* [online]. Seattle (Washington): Unearth Labs, c2022 [cit. 2023-03-22]. Dostupné z: <https://www.unearthlabs.com/blogs/gis-data-collection-tools>
- [10] OpenStreetMap Wiki contributors. GPS device reviews. *OpenStreetMap Wiki* [online]. OpenStreetMap Wiki [cit. 2023-03-22]. Dostupné z: https://wiki.openstreetmap.org/w/index.php?title=GPS_device_reviews&oldid=2376054

- [11] GPS Collection Devices: Selecting the Right GPS Receiver for Your Job. *D&D Larix, LLC – Resource Analysis* [online]. Pullman (Washington): D&D Larix, c2023 [cit. 2023-03-22]. Dostupné z: <http://resource-analysis.com/geospatial/gps-collection-devices/>
- [12] Mapové podklady: Základní mapový podklad. *Seznam Nápořěda* [online]. Praha: Seznam.cz, c1996-2023 [cit. 2023-04-21]. Dostupné z: <https://napoveda.seznam.cz/cz/mapy/mapove-podklady/zakladni-mapovy-podklad/>
- [13] Mapové podklady: Letecký mapový podklad. *Seznam Nápořěda* [online]. Praha: Seznam.cz, c1996-2023 [cit. 2023-04-21]. Dostupné z: <https://napoveda.seznam.cz/cz/mapy/mapove-podklady/letecky-mapovy-podklad/>
- [14] Nástroje: Import dat. *Seznam Nápořěda* [online]. Praha: Seznam.cz, c1996-2023 [cit. 2023-04-21]. Dostupné z: <https://napoveda.seznam.cz/cz/mapy/nastroje/import-dat/>
- [15] Mapy.cz. *Mapy.cz* [online]. Praha: Seznam.cz, c2023 [cit. 2023-04-21]. Dostupné z: <https://mapy.cz/>
- [16] RICHARD, Glenn, Diana KRUPNICK a Beth PRATT-SITULA. Teaching with Google Earth: What is Google Earth?. *Starting Point: Teaching Entry Level Geoscience* [online]. Northfield (Minnesota): SERC [cit. 2023-04-22]. Dostupné z: https://serc.carleton.edu/introgeo/google_earth/what.html
- [17] Import data from GPS devices. *Google Help* [online]. Mountain View (California): Google, c2023 [cit. 2023-04-22]. Dostupné z: <https://support.google.com/earth/answer/148095?hl=en>
- [18] SCHNEIDER, Adam. GPS Visualizer: Do-It-Yourself Mapping. *GPS Visualizer* [online]. Portland (Oregon): Schneider, c2003-2019 [cit. 2023-04-23]. Dostupné z: <https://www.gpsvisualizer.com/>
- [19] SCHNEIDER, Adam. About GPS Visualizer. *GPS Visualizer* [online]. Portland (Oregon): Schneider, c2003-2019 [cit. 2023-04-23]. Dostupné z: <https://www.gpsvisualizer.com/about.html>
- [20] SCHNEIDER, Adam. Make an HTML Map (Google/Leaflet) from a GPS file. *GPS Visualizer* [online]. Portland (Oregon): Schneider, c2003-2019 [cit. 2023-04-23]. Dostupné z: https://www.gpsvisualizer.com/map_input

- [21] SCHNEIDER, Adam. Draw a profile. *GPS Visualizer* [online]. Portland (Oregon): Schneider, c2003-2019 [cit. 2023-04-23]. Dostupné z: https://www.gpsvisualizer.com/profile_input
- [22] What is Arduino?: An introduction to what Arduino is, and what it can be used for. *Arduino Docs* [online]. Arduino, c2023 [cit. 2023-04-24]. Dostupné z: <https://docs.arduino.cc/learn/starting-guide/whats-arduino>
- [23] Arduino Hardware. *Arduino* [online]. Arduino, c2023 [cit. 2023-04-24]. Dostupné z: <https://www.arduino.cc/en/hardware>
- [24] SÖDERBY, Karl. Getting Started with Arduino: An introduction to hardware, software tools, and the Arduino API. *Arduino Docs* [online]. Arduino, c2023 [cit. 2023-04-25]. Dostupné z: <https://docs.arduino.cc/learn/starting-guide/getting-started-arduino>
- [25] JIMBLOM a EL DUDERINO. Arduino Shields v2. *SparkFun Electronics* [online]. Niwot (Colorado): SparkFun Electronics [cit. 2023-04-25]. Dostupné z: <https://learn.sparkfun.com/tutorials/arduino-shields-v2>
- [26] What is PlatformIO?: A place where Developers and Teams have true Freedom! No more vendor lock-in!. *PlatformIO* [online]. Tallinn: PlatformIO, c2014-2023 [cit. 2023-04-26]. Dostupné z: <https://docs.platformio.org/en/latest/what-is-platformio.html>
- [27] NEO-7 series: u-blox 7 GNSS modules. *U-blox* [online]. Thalwil (Švýcarsko): u-blox, c2015 [cit. 2023-05-02]. Dostupné z: https://content.u-blox.com/sites/default/files/products/documents/NEO-7_ProductSummary_%28UBX-13003342%29.pdf

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

GPIO	General-purpose input/output
GPS	Global Positioning System
GPX	GPS Exchange Format
I2C	Inter-Integrated Circuit
IDE	Integrated Development Environment
SD	Secure Digital
SPI	Seriál Peripheral Interface
UTC	Coordinated Universal Time

SEZNAM OBRÁZKŮ

Obrázek 1: Vybraná vývojová deska Arduino Mega 2560 (klon).....	22
Obrázek 2: Vybraný GPS modul u-blox NEO-7M.....	22
Obrázek 3: Vybraný SD modul	23
Obrázek 4: Vybraný modul OLED displej SH1106	23
Obrázek 5: Tlačítka na nepájivém poli	24
Obrázek 6: Blokové schéma zapojení hardwarové realizace.....	24
Obrázek 7: Hardwarová realizace	25
Obrázek 8: Blokové schéma fyzických modulů (obdélníky), softwarových modulů (zaoblené obdélníky) a datového toku mezi moduly (šipky)	26
Obrázek 9: Úvodní obrazovka zařízení	29
Obrázek 10: Základní rozložení hlavní obrazovky	29
Obrázek 11: Obrazovky vyplněná daty	29
Obrázek 12: Bitmapa <i>satellite</i>	32
Obrázek 13: Bitmapa <i>question</i>	32
Obrázek 14: Bitmapa <i>exclamation</i>	33
Obrázek 15: Bitmapa <i>splash</i>	33
Obrázek 16: Vygenerovaná kostra souboru.....	39
Obrázek 17: Vygenerovaný trasový bod	39
Obrázek 18: Uživatelské prostředí po inicializaci instance třídy	45
Obrázek 19: Dialogové okno pro zvolení souboru ve formátu GPX.....	46
Obrázek 20: Graf zobrazující trasu cesty autem mezi městy Uherský Brod a Zlín, vygenerován programem pro analýzu trasy uživatele. Data získána ze zařízení pro záznam trasy uživatele.....	47
Obrázek 21: Graf zobrazující ujetou vzdálenost trasy mezi městy Uherský Brod a Zlín, vygenerován programem pro analýzu trasy uživatele. Data získána ze zařízení pro záznam trasy uživatele.	48
Obrázek 22: Graf zobrazující rychlost pohybu zařízení na trase mezi městy Uherský Brod a Zlín, vygenerován programem pro analýzu trasy uživatele. Data získána ze zařízení pro záznam trasy uživatele.	49
Obrázek 23: Uživatelské prostředí s vygenerovanými informacemi o cestě mezi městy Uherský Brod a Zlín. Data získána ze zařízení pro záznam trasy uživatele.	50

SEZNAM PŘÍLOH

Příloha P1: CD-ROM obsahující zdrojový kód aplikací a ukázkové soubory vygenerované sledovacím zařízením