



## Zadání bakalářské práce

<b>Název:</b>	ETCS – DMI displej
<b>Student:</b>	Jan Stejskal
<b>Vedoucí:</b>	Ing. Jiří Chludil
<b>Studijní program:</b>	Informatika
<b>Obor / specializace:</b>	Webové a softwarové inženýrství, zaměření Softwarové inženýrství
<b>Katedra:</b>	Katedra softwarového inženýrství
<b>Platnost zadání:</b>	do konce letního semestru 2022/2023

### Pokyny pro vypracování

ETCS (European Train Control System) je jednotný celoevropský vlakový zabezpečovací systém. Cílem této práce je pro simulátor železničního vozidla vytvořit displej strojvedoucího (DMI – Driver Machine Interface).

1. Analyzujte dostupnou dokumentaci k ETCS (subset ERA-ERTMS-015560v2.3) s ohledem na fungování DMI.
2. Analyzujte diplomovou práci Petra Stříteského (FD ČVUT) zaměřenou na grafické knihovny pro DMI displej.
3. Na základě konzultací se zadavateli z Fakulty dopravní definujte funkční a nefunkční požadavky vybraných funkcionalit DMI displeje.
4. Pomocí metod softwarového inženýrství a s využitím externí grafické knihovny navrhnete vybrané funkcionality jednotlivých obrazovek displeje.
5. Implementujte vybrané funkcionality prototypu DMI displeje, prototyp musí být funkční na platformách Linux, Windows a na průmyslovém počítači dodaným zadavatelem.
6. Výsledný prototyp podrobte uživatelským, akceptačním a integračním testům.





**FAKULTA  
INFORMAČNÍCH  
TECHNOLÓGIÍ  
ČVUT V PRAZE**

Bakalářská práce

**ETCS – DMI displej**

*Jan Stejskal*

Katedra softwarového inženýrství

Vedoucí práce: Ing. Jiří Chludil

8. května 2022



---

## Poděkování

Rád bych poděkoval vedoucímu Ing. Jirímu Chludilovi za ochotný přístup, odborné vedení a cenné rady. Dále bych chtěl poděkovat mé rodině za podporu při tvorbě bakalářské práce i během studia.



---

# Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 2373 odst. 2 zákona č. 89/2012 Sb., občanský zákoník, ve znění pozdějších předpisů, tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu) licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 8. května 2022

.....

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2022 Jan Stejskal. Všechna práva vyhrazena.

*Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí a nad rámec oprávnění uvedených v Prohlášení na předchozí straně, je nezbytný souhlas autora.*

### **Odkaz na tuto práci**

Stejskal, Jan. *ETCS – DMI displej*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2022.



---

# Abstrakt

Tato práce se zabývá vývojem prototypu aplikace DMI displeje, který slouží jako ovládací displej strojvedoucího a zobrazuje potřebné informace. Aplikace je součástí projektu vlakového simulátoru s komponentou ETCS. V práci je analyzována nejen dokumentace určená pro tvorbu aplikace DMI displeje, ale také diplomová práce studenta Fakulty dopravní ČVUT, jehož knihovna, postavená na grafických knihovnách SDL 2.0, byla při implementaci využita. Cílem práce je návrh a implementace prototypu aplikace, který také komunikuje pomocí protokolu MQTT se softwarovým modulem EVC, což je další aplikace tvořená v rámci společného projektu vlakového simulátoru. Výsledný prototyp aplikace DMI je následně podroben uživatelským, akceptačním a integračním testům.

**Klíčová slova** DMI, ETCS, rozhraní člověk-stroj, průmyslový displej, drážní vozidlo, vlakový simulátor

---

# Abstract

This thesis deals with the development of a prototype DMI display application, which serves as a driver's control display and displays the necessary information. The application is part of a train simulator project with the ETCS component. The thesis analyzes not only the documentation intended for the creation of the DMI display application, but also the diploma thesis of a student of the Faculty of Transportation Sciences CTU, whose library, built on SDL 2.0 graphics libraries, was used in the implementation. The aim of the work is the design and implementation of a prototype application, which also communicates using the MQTT protocol with the EVC software module, which is another application created within a joint project of a train simulator. The resulting prototype of the DMI application is then subjected to usability, acceptance and integration tests.

**Keywords** DMI, ETCS, man-machine interface, industrial display, railway vehicle, train simulator

---

# Obsah

Úvod	1
<b>1 Cíl práce</b>	<b>3</b>
<b>2 Analýza dostupné dokumentace</b>	<b>5</b>
2.1 Rozsah platnosti dokumentace	5
2.2 Obecné zásady	5
2.2.1 Ohraničení objektů a vrstvy	5
2.2.2 Text	6
2.2.3 Symboly	6
2.2.4 Zvuky	6
2.3 Fyzické parametry	7
2.4 Uspořádání informací	7
2.4.1 Obrazovky	7
2.4.2 Tlačítka	7
2.5 Potvrzení	8
2.6 Jazyky	8
2.7 Rozložení displeje	9
2.8 Brzdné křivky – stav dohledu	9
2.9 Výchozí obrazovka	10
2.9.1 Rychlost a dohled	11
2.9.1.1 Rychloměr (Speed dial)	12
2.9.1.2 Kruhový ukazatel rychlosti (Circular speed gauge)	12
2.9.1.3 Brzdné informace	12
2.9.1.4 Doplnkové informace	13
2.9.2 Plánovací oblast	16
2.9.3 Monitorovací informace	17
2.9.4 Výběr tlačítek pro přepínání na podúrovňové obrazovky	18
2.10 Podúrovňové obrazovky – obecné vlastnosti	19

2.10.1	Proces zadávání/validace dat . . . . .	20
2.10.2	Přehled podúrovňových obrazovek . . . . .	21
2.11	Konkrétní podúrovňové obrazovky . . . . .	21
2.11.1	Menu obrazovky . . . . .	21
2.11.1.1	Hlavní obrazovka . . . . .	21
2.11.1.2	Obrazovka potlačení . . . . .	22
2.11.1.3	Speciální obrazovka . . . . .	23
2.11.1.4	Obrazovka nastavení . . . . .	24
2.11.2	Obrazovky pro zadávání dat . . . . .	25
2.11.2.1	Obrazovka čísla vlaku . . . . .	25
2.11.2.2	Obrazovka úrovně . . . . .	26
2.11.2.3	Obrazovka čísla strojvedoucího . . . . .	27
2.11.2.4	Obrazovka kontaktních údajů RBC . . . . .	28
2.11.2.5	Obrazovka jazyku . . . . .	29
2.11.2.6	Obrazovka hlasitosti . . . . .	30
2.11.2.7	Obrazovka jasu . . . . .	30
2.11.2.8	Obrazovka/y vlakových dat . . . . .	31
2.11.2.9	Obrazovka rychlosti/vzdálenosti v SR . . . . .	31
2.11.2.10	Obrazovka adheze . . . . .	32
2.11.3	Obrazovky ověření dat . . . . .	33
2.11.3.1	Obrazovka ověření vlakových dat . . . . .	33
2.11.4	Obrazovky prohlížení dat . . . . .	34
2.11.4.1	Obrazovka prohlížení dat . . . . .	34
2.12	Dialogové sekvence . . . . .	35
<b>3</b>	<b>Analýza diplomové práce Petra Stříteského</b>	<b>37</b>
3.1	Použité technologie . . . . .	37
3.2	Obsah knihovny . . . . .	37
3.2.1	Soubory . . . . .	37
3.2.2	Funkce . . . . .	38
3.3	Doporučené postupy pro použití knihovny . . . . .	38
3.3.1	Problém s knihovnou . . . . .	38
<b>4</b>	<b>Funkční a nefunkční požadavky</b>	<b>39</b>
4.1	Funkční požadavky . . . . .	39
4.1.1	Priorita funkčních požadavků . . . . .	42
4.2	Nefunkční požadavky . . . . .	44
<b>5</b>	<b>Návrh</b>	<b>45</b>
5.1	Komunikace s modulem EVC . . . . .	45
5.2	Zpracování formátu JSON . . . . .	46
5.3	Knihovna Petra Stříteského . . . . .	46
5.4	Architektura obrazovek . . . . .	46
5.5	Doménový model . . . . .	48

5.6	Důležité třídy . . . . .	49
5.7	Procesy . . . . .	51
5.7.1	Datový tok z modulu EVC do DMI . . . . .	51
5.7.2	Datový tok z DMI do modulu EVC . . . . .	52
5.7.3	Přepnutí obrazovky . . . . .	53
5.8	Přechodová mapa . . . . .	54
5.9	Změny oproti původnímu návrhu . . . . .	56
<b>6</b>	<b>Implementace</b>	<b>57</b>
6.1	Instalační příručka . . . . .	57
6.1.1	Windows . . . . .	57
6.1.2	Linux . . . . .	57
6.2	Vývojářská příručka . . . . .	58
6.2.1	Windows . . . . .	58
6.2.2	Linux . . . . .	62
6.3	Programátorská příručka . . . . .	63
6.4	Ukázka uživatelského rozhraní . . . . .	64
6.5	Uživatelská příručka . . . . .	66
6.5.1	Zahájení mise . . . . .	66
6.6	Schéma vlakového simulátoru . . . . .	69
<b>7</b>	<b>Testování</b>	<b>73</b>
7.1	Uživatelské testování . . . . .	73
7.1.1	Zadání čísla strojvedoucího a zahájení jízdy . . . . .	73
7.1.2	Zobrazení obrazovky prohlížení dat a změna čísla strojvedoucího . . . . .	74
7.1.3	Prohlížení plánovací oblasti . . . . .	75
7.2	Akceptační testy . . . . .	76
7.3	Integrační testy . . . . .	78
	<b>Závěr</b>	<b>79</b>
	<b>Literatura</b>	<b>81</b>
<b>A</b>	<b>Seznam použitých zkratk</b>	<b>83</b>
A.1	Zkratky k ETCS . . . . .	83
A.2	Ostatní . . . . .	84
<b>B</b>	<b>Obsah přiloženého CD</b>	<b>85</b>



---

## Seznam obrázků

2.1	Definice ohraničení, zdroj [1]	6
2.2	Rozložení displeje, zdroj [1]	9
2.3	Dohledové sekce, zdroj [1]	10
2.4	Výchozí obrazovka, zdroj [1]	11
2.5	Oblast rychlosti a dohledu, zdroj [1]	12
2.6	Indikátor zásahu nouzové/provozní brzdy, zdroj [1]	13
2.7	Informace o módu, zdroj [1]	14
2.8	Informace o úrovni, zdroj [1]	14
2.9	Informace o volné koleji vpřed, zdroj [1]	14
2.10	Textové zprávy, zdroj [1]	15
2.11	Příkazy a oznámení o stavu trati, zdroj [1]	15
2.12	Indikátor adhezního faktoru, zdroj [1]	15
2.13	Plánovací oblast, zdroj [1]	16
2.14	Indikace úspěšné komunikační relace, zdroj [1]	17
2.15	Indikace problému v komunikační relaci, zdroj [1]	18
2.16	Indikace povolení couvání, zdroj [1]	18
2.17	Symbol pro zobrazení geografické pozice, zdroj [1]	18
2.18	Tlačítka pro přepnutí na podúrovňové obrazovky, zdroj [1]	19
2.19	Hlavní obrazovka, zdroj [1]	22
2.20	Obrazovka potlačení, zdroj [1]	23
2.21	Speciální obrazovka, zdroj [1]	24
2.22	Obrazovka nastavení, zdroj [1]	25
2.23	Obrazovka čísla vlaku, zdroj [1]	26
2.24	Obrazovka úrovně, zdroj [1]	27
2.25	Obrazovka čísla strojvedoucího, zdroj [1]	28
2.26	Obrazovka kontaktních údajů RBC, zdroj [1]	29
2.27	Obrazovka jazyku, zdroj [1]	30
2.28	Obrazovka vlakových dat, zdroj [1]	31
2.29	Obrazovka rychlosti/vzdálenosti v SR, zdroj [1]	32
2.30	Obrazovka adheze, zdroj [1]	33

2.31	Obrazovka ověření vlakových dat, zdroj [1]	34
2.32	Obrazovka prohlížení dat, zdroj [1]	35
5.1	Doménový model	48
5.2	Komunikace z EVC do DMI	51
5.3	Komunikace z DMI do EVC	52
5.4	Přepnutí obrazovek	53
5.5	Přechodová mapa obrazovek	55
6.1	Průzkumník řešení	59
6.2	Karta vlastností souboru	60
6.3	Konfigurace projektu	60
6.4	Okno vlastností projektu	61
6.5	Úprava proměnných prostředí	62
6.6	Adresářový diagram	64
6.7	Jízda za normálních podmínek	65
6.8	Jízda v kritické situaci	66
6.9	Obrazovka čísla strojvedoucího	67
6.10	Hlavní obrazovka	68
6.11	Schéma ETCS	69



---

## Seznam tabulek

2.1	Tabulka podúrovňových obrazovek . . . . .	21
4.1	Tabulka priority funkčních požadavků . . . . .	43
7.1	Tabulka schválených funkčních požadavků . . . . .	77



---

# Úvod

Projekt vlakového simulátoru s komponentou celoevropského vlakového zabezpečovacího systému vznikl v rámci spolupráce Fakulty dopravní a Fakulty informačních technologií Českého vysokého učení technického v Praze. Cílem projektu je vytvoření prototypu simulátoru na Fakultě dopravní ČVUT, který by umožňoval výcvik strojvedoucích v ČR. Tento projekt začal v minulém akademickém roce v rámci předmětu Softwarový týmový projekt 1 na Fakultě informačních technologií ČVUT a nadále pokračuje. Aplikace DMI displeje, o které tato práce pojednává, funguje jako ovládací displej strojvedoucího a je jedním z modulů komponenty ETCS.

S projektem jsem se setkal při jeho začátcích v rámci předmětu Softwarový týmový projekt 1, kdy jsem si zvolil účast na aplikaci DMI displeje. Na konci předmětu byla v rámci implementace všem zájemcům z řad účastníků projektu nabídnuta možnost letní prázdninové brigády, během které jsem se podílel na programování určitých funkcionalit aplikace společně s dalšími třemi kolegy. V následujícím zimním semestru se poté naskytl možnost na aplikaci dále pokračovat v rámci bakalářské práce, což mě velmi zaujalo, jelikož mě práce na tomto projektu bavila. Díky tomu jsem měl vybrané téma mé bakalářské práce, jejímž cílem je vytvoření prototypu aplikace DMI displeje.

Práce je rozdělena do šesti kapitol, které pokrývají analýzu, funkční a nefunkční požadavky, návrh, implementaci a testování aplikace.

Analytická část se zabývá dostupnou dokumentací pro tvorbu aplikace DMI displeje, poskytující konkrétní pokyny k chování a vzhledu, a také diplomovou prací studenta Fakulty dopravní ČVUT, jejímž výstupem je grafická knihovna pro tvorbu displejů drážních vozidel, která je použita v implementaci aplikace.

Z analýzy dokumentace a konzultace se zadavateli z Fakulty dopravní ČVUT vzešly poté konkrétní funkční a nefunkční požadavky.

V návrhu je popsána aktuální architektura aplikace, důležité procesy a použité technologie.

## Úvod

---

Implementační část této práce prezentuje zejména příručky určené pro instalaci aplikace, její použití a také pro vývoj.

Poslední kapitola se věnuje testování prototypu aplikace. Pro tento účel jsou zde rozepsány testovací scénáře v rámci uživatelského testování, dále pak akceptační a integrační testy.

---

## Cíl práce

Cílem této bakalářské práce je vytvoření prototypu aplikace DMI displeje, který bude funkční na operačních systémech Windows, Linux a také na průmyslovém displeji poskytnutém od zadavatelů z Fakulty dopravní ČVUT. Cílem dílčích částí práce je analyzovat potřebnou dostupnou dokumentaci pro tvorbu aplikace a také diplomovou práci, která se zabývá tvorbou displejů drážních vozidel. Dále je nutné stanovit konkrétní funkční a nefunkční požadavky, na jejichž základě bude prototyp aplikace DMI navrhnout a implementován. Výsledný prototyp bude podroben uživatelským, akceptačním a integračním testům. Práce bude poté sloužit také jako příručka pro uživatele a zejména pak pro budoucí vývojáře, kteří se budou podílet na dalším vývoji této aplikace.



---

# Analýza dostupné dokumentace

Tato kapitola se zaměřuje na analýzu dostupné dokumentace k systému ETCS se zaměřením na DMI displej ve specifikační verzi 2.3.0, což představuje dokument s názvem ERA-ERTMS-015560v2.3, který je hlavním zdrojem pokynů a pravidel pro aplikaci DMI displeje. Obrázky, které jsou převzaty z dokumentu ve formě screenshotů, jsou upravené v grafickém editoru z důvodu lepší viditelnosti a snazšího odlišení objektů na obrázku.

## 2.1 Rozsah platnosti dokumentace

Dokumentace popisuje rozhraní mezi strojvedoucím a systémem ETCS. Poskytuje definice k informacím, které představují odezvu na provozní situace. Tyto definice zahrnují vizuální informace ve formě symbolů a textových zpráv, jejich umístění a také zvukové doprovody. Interakce strojvedoucího s displejem je upřesněna pomocí názorných diagramů. Viz kapitola 3 v [1].

Specifikace umožňuje tvorbu funkcionalit DMI displeje pomocí 2 technologií, konkrétně dotykové obrazovky a soft tlačítek. Viz kapitola 3 v [1]. Tato práce se ale bude zabývat pouze řešením pro dotykovou obrazovku.

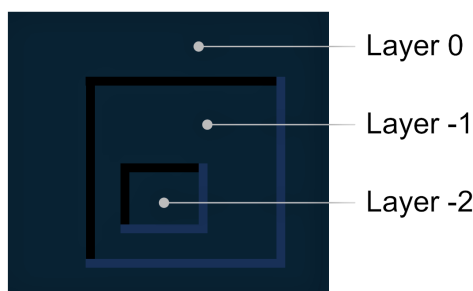
## 2.2 Obecné zásady

Pokud není v dokumentu uvedeno jinak, jsou všechny konkrétní číselné údaje o rozměrech objektů uvedeny ve speciální základní jednotce, tzv. cells (buňky), která se v závislosti na použitém rozlišení aplikace může skládat z jednoho, nebo více pixelů. Viz kapitola 4.2 v [1].

### 2.2.1 Ohraničení objektů a vrstvy

Dokumentace poskytuje informace o způsobu ohraničení kolem konkrétních typů objektů, vytvářející vizuální efekt trojrozměrného vzhledu, který objekt prohlubuje, nebo naopak vystupuje. Těchto dojmů lze docílit pomocí použití

specifických barev a vykreslení jednoho, resp. dvou okrajů, které se poté vrství. Pro snazší pochopení je uveden i popis konkrétních vrstev, označených čísly 0, -1, -2, a jejich vzájemné propojení. Viz kapitola 5.1 v [1].



Obrázek 2.1: Definice ohraničení, zdroj [1]

Pokud je vyžadované potvrzení objektu od strojvedoucího, je okolo něj použit speciální případ žlutého blikajícího rámečku s frekvencí 2 Hz. Viz kapitola 5.1 v [1].

### 2.2.2 Text

Dokumentace popisuje důležité parametry pro použité písmo, které musí být bezpatkové, proporcionální a s výškou vztahenou k velkým znakům. Konkrétní výška v číslech se poté liší v závislosti na oblasti, ve které je písmo použité. Doporučené fonty jsou Helvetica, Verdana, Swiss, and Chicago. Viz kapitola 5.1 v [1].

Dále je definováno i odsazení textu a jeho základní barva (šedá), která se v případě potřeby konkrétní oblasti může měnit. Pravidla pro zobrazování numerických či alfanumerických dat jsou vztahena k seskupování znaků a oddělování těchto skupin pro lepší přehlednost a jednodušší čitelnost souvislých dat. Viz kapitola 5.1 v [1].

### 2.2.3 Symboly

Pro zpřehlednění orientace mezi symboly použitými v aplikaci, je na konci dokumentace umístěná informační tabulka zahrnující všechny tyto symboly společně s jejich umístěním, rozměry, stručným popisem a názvem příslušného souboru. Symboly představují obrázky ve formátu *.bmp*. Viz kapitola 11 v [1].

### 2.2.4 Zvuky

Dokumentace specifikuje konkrétní případy použití celkem čtyř zvuků včetně délky přehrávání. První zvuk signalizuje zmáčknutí tlačítka, druhý se týká obecného upozornění a zbylé dva se týkají monitoringu rychlosti. Všechny položky jsou ve formátu *.wav*. Viz kapitola 12 v [1].



## 2.3 Fyzické parametry

### Obecné parametry

Dle specifikací dokumentace musí být minimální rozměry celkové zobrazované oblasti 180 mm \* 135 mm (šířka \* výška), přičemž minimální požadované rozlišení celkové zobrazované oblasti je 640 \* 480 buněk. Viz kapitola 5.2 v [1].

### Barvy

V dokumentaci jsou upřesněny konkrétní RGB hodnoty 24-bitových barev použitých v aplikaci. Pro přehlednění slouží informační tabulka se souhrnem barev a hodnot jejich složek. Viz kapitola 5.2 v [1].

### Jas

Úpravy úrovně jasu aplikace provádí řidič, případně jsou provedeny automaticky, přičemž při zapnutí aplikace se použije poslední zvolená hodnota, nebo pokud není žádná taková hodnota uložena, použije se medián rozsahu hodnot, které ale nejsou specifikovány v tomto dokumentu. Viz kapitola 5.2 v [1].

### Hlasitost

Řidič má možnost upravovat úroveň hlasitosti. Při zapnutí aplikace se stejným způsobem jako v případě jasu použije poslední zvolená hodnota, nebo pokud není žádná taková hodnota uložena, je použit medián rozsahu hodnot, které ale také nejsou specifikovány v tomto dokumentu. Viz kapitola 5.2 v [1].

## 2.4 Uspořádání informací

### 2.4.1 Obrazovky

Obrazovky slouží pro prezentaci objektů, textových zpráv a tlačítek alokovaných v konkrétních oblastech. Tato kapitola v dokumentaci se zaměřuje na obecný popis částí, ze kterých by se měly obrazovky skládat, a také na deskripci vztahu pro zobrazování a překrývání obrazovek. Viz kapitola 5.3.1 v [1].

### 2.4.2 Tlačítka

Tlačítka se vždy nacházejí v jednom ze 3 stavů, nazvaných „povolené“, „vypnuté“, „stisknuté“. Jejich stisknutí je poté doprovázeno vizuálním a hmatovým nebo zvukovým doprovodem. Všechna tlačítka jsou poté jedním ze 4 typů:

- Up-type – K aktivaci tlačítka dochází až po uvolnění stlačení.
- Down-type – K aktivaci dochází okamžitě po stisknutí.
- Down-type s funkcí opakování – K aktivaci dojde po stisknutí, a pokud je i nadále stlačeno po dobu 1,5 sekundy, zapne se funkce opakování, kdy k aktivaci dochází každých 0,3 sekundy, než je tlačítko uvolněno.
- Delay-type – Po dobu stlačení se po každých 0,5 sekundách mění stav tlačítka mezi „stisknuté“ a „povolené“, naččež pokud po 2 sekundách od prvního stisknutí tlačítka je stále stlačeno, dojde k aktivaci až po uvolnění stlačení.

Dokumentace dále poskytuje informace k velikosti dotykové plochy tlačítek, barvě pozadí a také k pravidlům chování všech jednotlivých zmíněných typů tlačítek. Viz kapitola 5.3.2 v [1].

### 2.5 Potvrzení

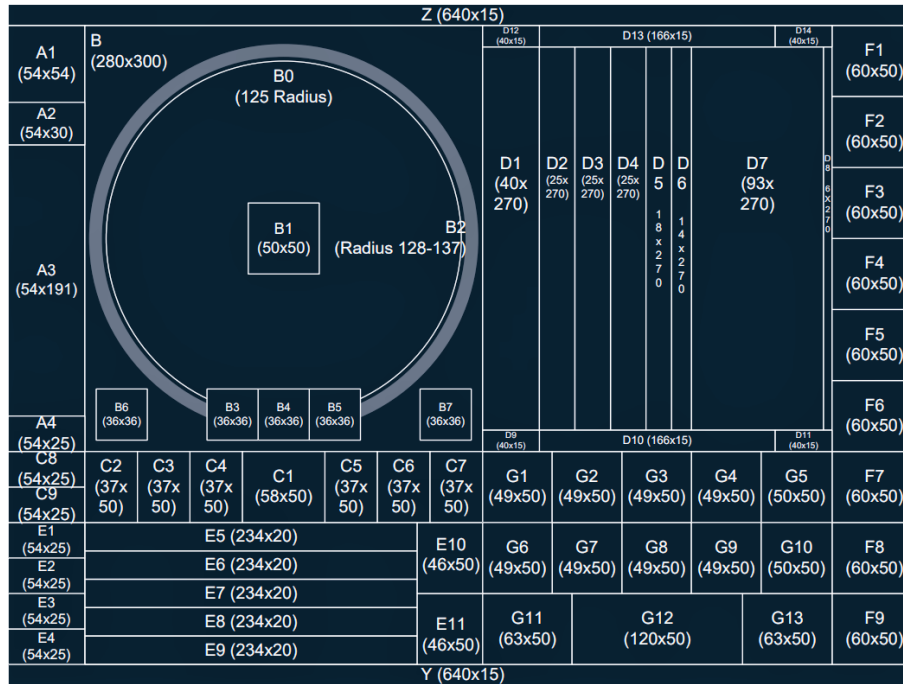
V určitých situacích je nutné, aby strojvedoucí potvrdil potřebnou informaci v aplikaci ve formě speciálního tlačítka typu „up-type“ se žlutým blikajícím rámečkem, které po stisknutí zmizí. Lokace tohoto tlačítka je závislá na lokaci potvrzovaného objektu, kterým může být například nějaký symbol, nebo text. Na obrazovce může být zobrazeno vždy pouze jedno upozornění pro potvrzení, nikoli více najednou. Viz kapitola 5.4 v [1].

### 2.6 Jazyky

Aplikace by měla být schopna zobrazit všechny text ve specifikovaných jazycích, přičemž výběr konkrétního jazyku by měl být umožněn v obrazovce s nastavením nebo v případě zapnutí aplikace se použije poslední zvolený. Viz kapitola 5.5 v [1].

## 2.7 Rozložení displeje

Displej sestává celkem z 9 hlavních oblastí, které se dále dělí na jednotlivé podoblasti. Pozice jednotlivých oblastí jsou vždy vztaženy k levému hornímu rohu. Právě na tyto dílčí části se poté v následujících kapitolách dokumentace velmi často odkazuje ve spojení s různými objekty. Viz kapitola 6 v [1].

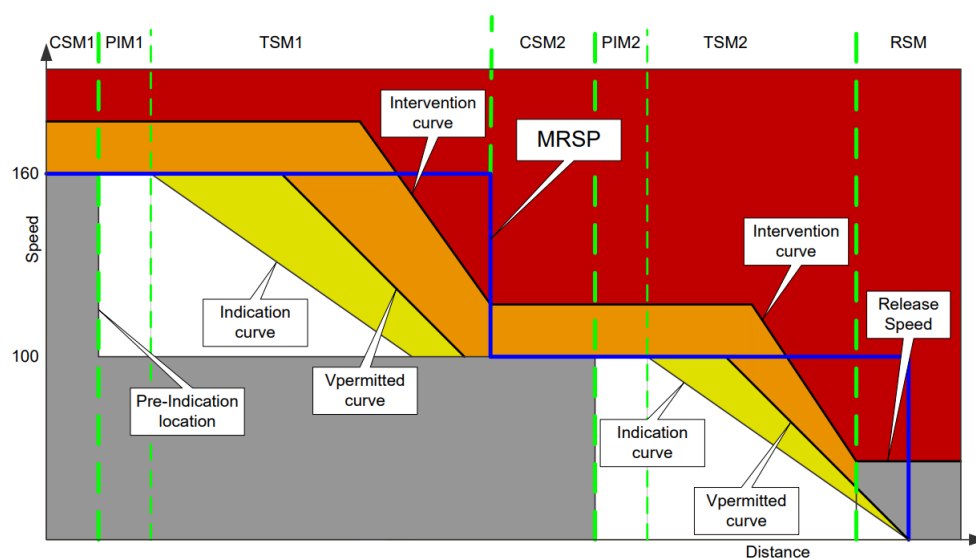


Obrázek 2.2: Rozložení displeje, zdroj [1]

## 2.8 Brzdné křivky – stav dohledu

Dokumentace seznamuje se stavy dohledu, které monitorují rychlost vlaku a jsou založené na brzdných křivkách. Některé konkrétní objekty, nebo jejich dílčí části (např. rychloměr, kruhový ukazatel rychlosti) na základě těchto stavů mění svou barvu. Viz kapitola 7 v [1].

Pro znázornění se zde nachází i obrázek s celkovým přehledem vztaženým ke stavům dohledu. Konkrétní podmínky zbarvení objektů definuje kapitola 8 v [1].



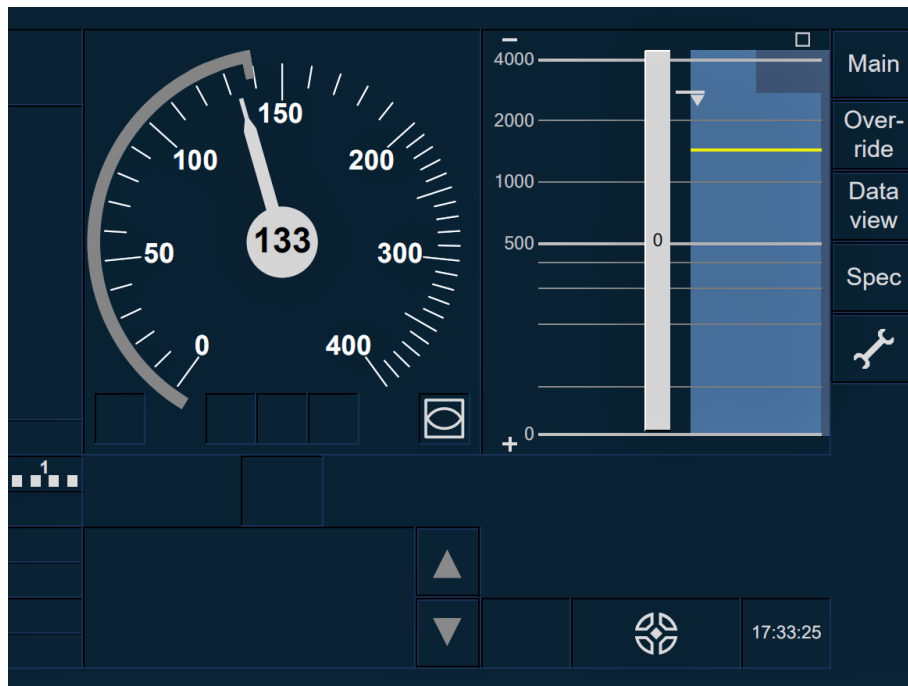
Obrázek 2.3: Dohledové sekce, zdroj [1]

Obrázek je rozdělen na následující dohledové sekce, které se poté zvlášť rozdělují na konkrétní stavy dohledu, viz kapitola 7 v [1]:

- Dvě sekce monitorování stropní rychlosti (CSM)
- Dvě preindikační monitorovací sekce (PIM)
- Dvě sekce monitorování cílové rychlosti (TSM)
- Jedna sekce monitorování uvolňovací rychlosti (RSM)

### 2.9 Výchozí obrazovka

Výchozí obrazovka je hlavním informačním zdrojem, který strojvedoucímu prezentuje data o monitorování rychlosti vlaku, brzdných křivkách, plánovací oblasti, textových zprávách a další nezbytné informace ohledně jízdy a řízení vlaku. Výchozí obrazovka zaujímá celou plochu rozložení displeje. V této kapitole se zaměřím na její dílčí části. Viz kapitola 8 v [1].



Obrázek 2.4: Výchozí obrazovka, zdroj [1]

### 2.9.1 Rychlost a dohled

Sekce rychlosti a dohledu se zabývá objekty, které umožňují zobrazit informace vztahující se k rychlosti a stavům dohledu. Následující obrázek popisuje některé objekty z této sekce, přičemž podkapitoly, které se jimi zabývají, mají v závorce i svůj anglický přepis z důvodu orientace v tomto obrázku. Viz kapitola 8.2 v [1].



Obrázek 2.5: Oblast rychlosti a dohledu, zdroj [1]

### 2.9.1.1 Rychloměr (Speed dial)

Pro zobrazení rychlosti vlaku slouží rychloměr s ručičkou. Rychlostní stupnice je jedním z následujících čtyř typů dle vyžadované konfigurace, viz kapitola 8.2.1 v [1]:

- 0 km/h až 400 km/h
- 0 km/h až 250 km/h
- 0 km/h až 180 km/h
- 0 km/h až 140 km/h

### 2.9.1.2 Kruhový ukazatel rychlosti (Circular speed gauge)

Kruhový ukazatel rychlosti je křivka umístěná okolo rychloměru, sloužící pro názorné zobrazení povolené rychlosti, cílové rychlosti, zásahové rychlosti a rychlosti uvolnění, v závislosti na aktuálním stavu dohledu, který se aktivuje vždy na základě brzdnych křivek. Viz kapitola 8.2.1 v [1].

Jeho další vlastností je, dle vztahu aktuální rychlosti vlaku k výše zmíněným rychlostem, reakce ve formě změny barvy křivky v určitých částech. Dokumentace obsahuje pro zpřehlednění těchto barevných změn tabulku vztahu stavů dohledu k barvám jednotlivých částí kruhového ukazatele. Viz kapitola 8.2.1 v [1].

### 2.9.1.3 Brzdné informace

Brzdné informace tvoří vzdálenost k cíli a indikátor zásahu nouzové/provozní brzdy. Viz kapitola 8.2.2 v [1].

### Vzdálenost k cíli (Target distance)

Vzdálenost k cíli, která představuje vzdálenost k nejbližšímu snížení povolené rychlosti vlaku, je reprezentována logaritmicky škálovaným svislým pruhem se stupnicí a číselným údajem vzdálenosti.

Vzdálenost, kterou stupnice se svislým pruhem znázorňuje je 0 až 1000 metrů. Dokumentace dále zpřesňuje, jakým způsobem a kdy se tento pruh vykresluje. Viz kapitola 8.2.2 v [1].

### Indikátor zásahu nouzové/provozní brzdy

Indikátor zásahu nouzové/provozní brzdy je zastoupen patřičným symbolem, který se vyobrazí a zároveň může posloužit jako tlačítko pro potvrzení. Viz kapitola 8.2.2 v [1].



Obrázek 2.6: Indikátor zásahu nouzové/provozní brzdy, zdroj [1]

### Funkce vypnutí/zapnutí zobrazení informací o rychlosti

Dokumentace definuje podmínky, za kterých je možné vypnout/zapnout zobrazení určitých informací, týkajících se rychlosti vlaku. Viz kapitola 8.2.2 v [1].

#### 2.9.1.4 Doplnkové informace

Doplňkovými informacemi se rozumí informace o módu, úrovni, volné koleji před vlakem, dále také textové zprávy, příkazy a oznámení o stavu trati a indikátor adhezního faktoru. Viz kapitola 8.2.3 v [1].

### Informace o módu (Mode indication)

Použitím příslušných symbolů se znázorňují, dle upřesněných pravidel, následující tři elementy, tvořící informace o módu, viz kapitola 8.2.3 v [1]:

- Aktuální ERTMS/ETCS mód
- Případné požadované potvrzení módu
- Zobrazení symbolu potlačení z důvodu informování řidiče o aktivní funkci potlačení



Obrázek 2.7: Informace o módu, zdroj [1]

### Informace o úrovni (Level indication)

Dokumentace specifikuje pravidla pro zobrazení následujících tří složek, které formují informace o úrovni, viz kapitola 8.2.3 v [1]:

- Aktuální ERTMS/ETCS úroveň
- Oznámení a potvrzení ERTMS/ETCS úrovně
- Oznámení ERTMS/ETCS úrovně



Obrázek 2.8: Informace o úrovni, zdroj [1]

### Informace o volné koleji před vlakem

Informace o volné koleji před vlakem se zobrazuje pomocí příslušného symbolu a tlačítka, které vyžaduje potvrzení od strojvedoucího. Viz kapitola 8.2.3 v [1].

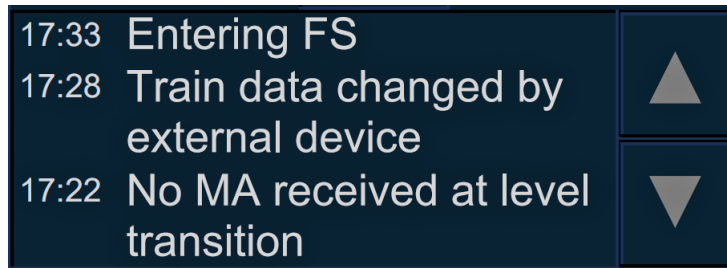


Obrázek 2.9: Informace o volné koleji vpřed, zdroj [1]



### Textové zprávy

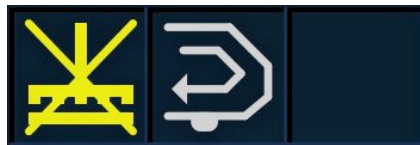
Aplikace zobrazuje textové zprávy obdržené od trati a operační textové zprávy specifikované v kapitole 13 v [1] a v novém trakčním systému. Některé ze zpráv je nutné potvrdit strojevedoucím. Viz kapitola 8.2.3 v [1].



Obrázek 2.10: Textové zprávy, zdroj [1]

### Příkazy a oznámení o stavu trati (Orders and Announcements)

S využitím konkrétních symbolů dle dokumentace se v aplikaci zobrazují příkazy a oznámení o stavu trati. Viz kapitola 8.2.3 v [1].



Obrázek 2.11: Příkazy a oznámení o stavu trati, zdroj [1]

### Indikátor adhezního faktoru

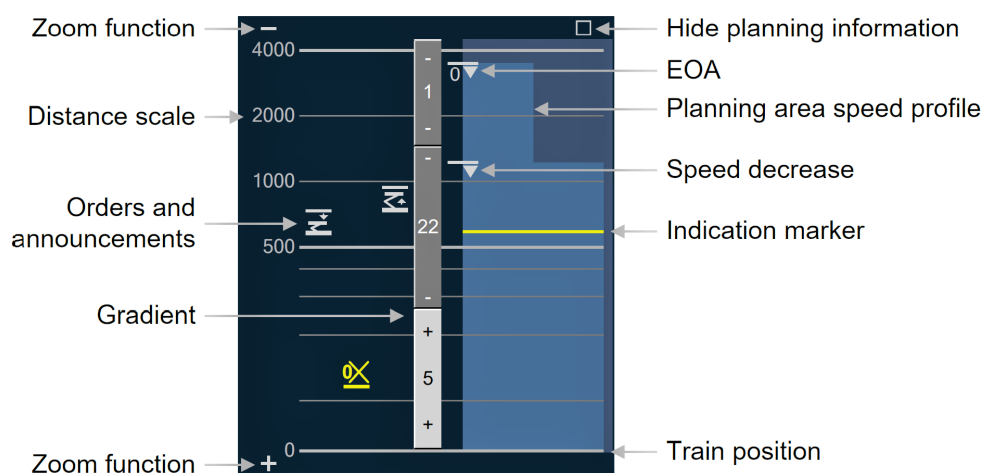
V aplikaci se v případě kluzké trati vyobrazí příslušný symbol. Viz kapitola 8.2.3 v [1].



Obrázek 2.12: Indikátor adhezního faktoru, zdroj [1]

### 2.9.2 Plánovací oblast

Plánovací oblast zobrazuje strojvedoucímu informace o podmínkách a dění před vlakem. Následující obrázek znázorňuje všechny objekty vyskytující se v plánovací oblasti. Viz kapitola 8.3 v [1].



Obrázek 2.13: Plánovací oblast, zdroj [1]

#### Stupnice vzdálenosti (Distance scale)

Tento objekt zobrazuje logaritmičticky škálovanou stupnici vzdálenosti, která je vykreslena ve formě přímk s příslušnými číselnými vzdálenostmi. Viz kapitola 8.3.2 v [1].

#### Příkazy a oznámení (Orders and announcements)

Ve formě symbolů jsou znázorněny příkazy a oznámení na trati. Viz kapitola 8.3.3 v [1].

#### Profil gradientů (Gradient)

V plánovací oblasti se vykresluje také profil gradientů, které se nacházejí před vlakem, ve formě vertikálně orientovaných obdélníků, znázorňujících sklony na trati. Viz kapitola 8.3.4 v [1].

#### Změny rychlostního profilu (Speed decrease + EOA)

Tento objekt vyobrazuje symboly, které popisují, zdali v oblasti nejrestriktivnější rychlosti vlaku dochází ke změnám rychlosti, tedy jejímu zvýšení nebo snížení. Viz kapitola 8.3.5 v [1].

### Rychlostní profil plánovací oblasti (Planning area speed profile)

Rychlostní profil plánovací oblasti je vykreslen ve formě diagramu vztahu rychlosti a vzdálenosti. Horizontální osa znázorňuje rychlostní stupnice a vertikální osa zobrazuje nespojitosti na základě vzdálenostní stupnice. Viz kapitola 8.3.6 v [1].

### Indikační značka (Indication marker)

Jedná se o horizontální žlutě zbarvenou čáru, jejíž lokace je relativní k indikační vzdálenosti aktuálního cíle. Viz kapitola 8.3.7 v [1].

### Funkce přiblížení/oddálení a schování/zobrazení

Plánovací oblast obsahuje navíc dvě tlačítka pro přiblížení a oddálení s příslušnými znaménky „+“ a „-“, které po kliknutí zobrazí náležitou stupnici vzdálenosti, reprezentující přiblížení, resp. oddálení. Poslední funkcí v této oblasti je pak možnost schování a zobrazení plánovacích informací pomocí tlačítka. Viz kapitola 8.3 v [1].

## 2.9.3 Monitorovací informace

Jedná se o další doplňkové informace pro strojvedoucího, které musí aplikace zobrazovat. Viz kapitola 8.4 v [1].

### Indikace komunikační relace

Indikace komunikační relace se zobrazí ve formě příslušného symbolu ve chvíli, kdy do DMI přijde zpráva o navázání spojení s modulem RBC. Viz kapitola 8.4.1 v [1].



Obrázek 2.14: Indikace úspěšné komunikační relace, zdroj [1]

V případě, že navázání komunikace bylo neúspěšné nebo došlo k náhlému přerušení fungujícího spojení, zobrazí se ihned na 20 sekund symbol, který reprezentuje tyto podmínky. Viz kapitola 8.4.1 v [1].



Obrázek 2.15: Indikace problému v komunikační relaci, zdroj [1]

### Indikace povolení couvání

V případě, že vlak stojí v oblasti couvání, zobrazí se symbol, který značí povolení pro couvání. Viz kapitola 8.4.2 v [1].



Obrázek 2.16: Indikace povolení couvání, zdroj [1]

### Lokální čas

Aplikace zobrazuje lokální čas ve formátu „HH:MM:SS“. Viz kapitola 8.4.3 v [1].

### Geografická pozice

Aplikace zobrazuje aktuální geografickou pozici vlaku. Symbol funguje zároveň jako „down-type“ tlačítko pro zapnutí a vypnutí zobrazení geografické pozice vlaku. Viz kapitola 8.4.4 v [1].



Obrázek 2.17: Symbol pro zobrazení geografické pozice, zdroj [1]

## 2.9.4 Výběr tlačítek pro přepínání na podúrovňové obrazovky

Tato selekce zahrnuje výběr pěti tlačítek typu „up-type“, které umožňují strojvedoucímu přepnutí na následující podúrovňové obrazovky: Hlavní (Main), Potlačení (Override), Prohlížení dat (Data view), Speciální (Special), Nastavení (Settings). Viz kapitola 8.5 v [1].



Obrázek 2.18: Tlačítka pro přepnutí na podúrovňové obrazovky, zdroj [1]

## 2.10 Podúrovňové obrazovky – obecné vlastnosti

Podúrovňové obrazovky jsou takové obrazovky, které překrývají pravou polovinu výchozí obrazovky, nebo jsou rozloženy napříč celou zobrazovací plochou. Každá obrazovka obsahuje tlačítko pro přepnutí zpět, typu „up-type“. Dokumentace upřesňuje rozměry a pozice obsažených objektů.

Následující podkapitoly se zaměřují na jednotlivé typy podúrovňových obrazovek. Viz kapitola 9 v [1].

### Menu obrazovky

Jedná se o obrazovky, jejichž úkolem je prezentovat strojvedoucímu selekci tlačítek, které spouštějí konkrétní funkcionalitu, nebo přepínají na jinou obrazovku. Tato tlačítka jsou typu „up-type“, pokud není definováno jinak. Viz kapitola 9.2 v [1].

### Obrazovky pro zadávání dat

Tento typ obrazovek umožňuje zadávání hodnot do datových typů. Pro tento účel se využívají následující objekty, které obrazovka obsahuje. Viz kapitola 9.3 v [1].

### **Pole pro vstup**

Pole pro vstup je objekt obdélníkového tvaru, do kterého se zadávají data. Pro indikaci místa vložení následujícího znaku slouží kurzor ve formě blikajícího podtržítka s frekvencí 2 Hz. Dokumentace pak dále definuje pravidla pro zarovnání zadaných dat a jejich kontrolu. Těchto polí může obrazovka obsahovat několik najednou, maximálně však čtyři. Viz kapitola 9.3 v [1].

### **Echo text**

Jedná se o textový objekt, jehož účelem je zobrazovat hodnotu dat zadaných patřičným polem pro vstup. Výskyt echo textů je ale podmíněn přítomností více než jednoho pole pro vstup a jejich celkový počet je roven počtu dat, které se pomocí dané obrazovky zadávají. Viz kapitola 9.3 v [1].

### **Klávesnice**

Pro potřeby aplikace existují tyto čtyři druhy klávesnic: numerická, vylepšená numerická, alfanumerická a dedikovaná s předdefinovanými volbami. V případě jednoho pole pro vstup se vždy zobrazuje pouze jedna klávesnice, ale pokud je polí pro vstup více, pak na základě zvoleného pole se zobrazí s ním spojená příslušná klávesnice. Viz kapitola 9.3 v [1].

### **Obrazovky pro ověření dat**

Tyto obrazovky se skládají z jednoho pole pro vstup, echo textů příslušných dat k validaci a také dedikované klávesnice, která obsahuje pouze dvě tlačítka: Ano, Ne. Účelem této obrazovky je ověření, zdali zadané hodnoty dat jsou správné. Viz kapitola 9.4 v [1].

### **Obrazovky prohlížení dat**

Účelem obrazovek tohoto typu je zobrazení určitých dat. Prezentovaná data se zobrazují pomocí štítků s označením typu dat a vedle nich zarovnané samotné hodnoty, které se ale zobrazují pouze v případě, že jsou validní. Dokumentace také udává pravidla pro zarovnání textů a způsob jejich zobrazení. Viz kapitola 9.5 v [1].

#### **2.10.1 Proces zadávání/validace dat**

Proces zadávání/validace konkrétních dat by se měl zahájit, když dojde k zobrazení první obrazovky pro zadávání dat, která se vztahuje k těmto konkrétním datům.

V případě, že neexistuje obrazovka pro ověření zadávaných dat, pak je proces zadávání/validace dat ukončen akceptováním zadaných dat pomocí

tlačítka s nápisem „Ano“ nebo opuštěním obrazovky pomocí tlačítka pro zavření obrazovky.

Pokud existuje obrazovka pro ověření zadávaných dat, pak je tento proces ukončen, když v ní strojvedoucí stiskne tlačítko s nápisem „Ano“, čímž potvrdí vložené hodnoty, dále pak pokud je obrazovka pro zadávání těchto dat nebo obrazovka pro ověření těchto dat opuštěna pomocí tlačítka pro zavření příslušné obrazovky. Viz kapitola 9.6 v [1].

### 2.10.2 Přehled podúrovňových obrazovek

Tabulka 2.1: Tabulka podúrovňových obrazovek

Typ	Obrazovky
Menu	Hlavní, Potlačení, Speciální, Nastavení
Zadávaní dat	Číslo vlaku, Úroveň, Číslo řidiče, kontaktní údaje RBC, Jazyk, Hlasitost, Jas, Vlaková data, Rychlost/vzdálenost v SR, Adheze
Ověření dat	Ověření vlakových dat
Prohlížení dat	Prohlížení dat

## 2.11 Konkrétní podúrovňové obrazovky

V této kapitole se nachází popis všech konkrétních podúrovňových obrazovek, které aplikace zahrnuje.

### 2.11.1 Menu obrazovky

#### 2.11.1.1 Hlavní obrazovka

Hlavní obrazovka prezentuje strojvedoucímu výběr sedmi tlačítek. Všechna tlačítka jsou typu „up-type“, kromě „Posun“ a „Nikoliv vedoucí“, které jsou obě „delay-type“, ale po stisknutí se změjí na „up-type“, přičemž další stlačení je vrátí zpět na „delay-type“. Funkce tlačítek jsou následující, viz kapitola 10.2.1 v [1]:

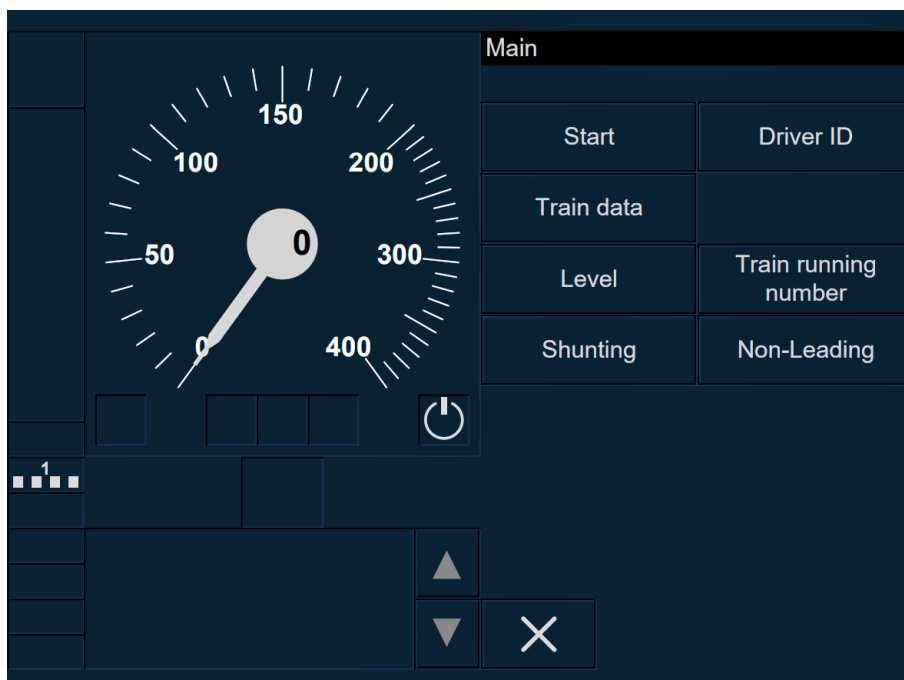
- Start – Pro obdržení oprávnění k jízdě nebo autorizace odpovědnosti strojvedoucího.
- Číslo řidiče (Driver ID) – Pro přepnutí na obrazovku čísla řidiče.
- Vlaková data (Train data) – Pro přepnutí na obrazovku vlakových dat.
- Úroveň (Level) – Pro přepnutí na obrazovku úrovně.

## 2. ANALÝZA DOSTUPNÉ DOKUMENTACE

---

- Číslo vlaku (Train running number) – Pro přepnutí na obrazovku čísla vlaku.
- Posun (Shunting) – Pro spuštění procedury „Posun“.
- Nikoliv vedoucí (Non-Leading) – Pro spuštění procedury „Nikoliv vedoucí“.
- Zpět – Pro návrat do výchozí obrazovky.

Pro podrobné vysvětlení chování aplikace po stisknutí příslušného tlačítka za specifických podmínek je v kapitole 10.7.3 v [1] uvedena tabulka a názorný diagram.

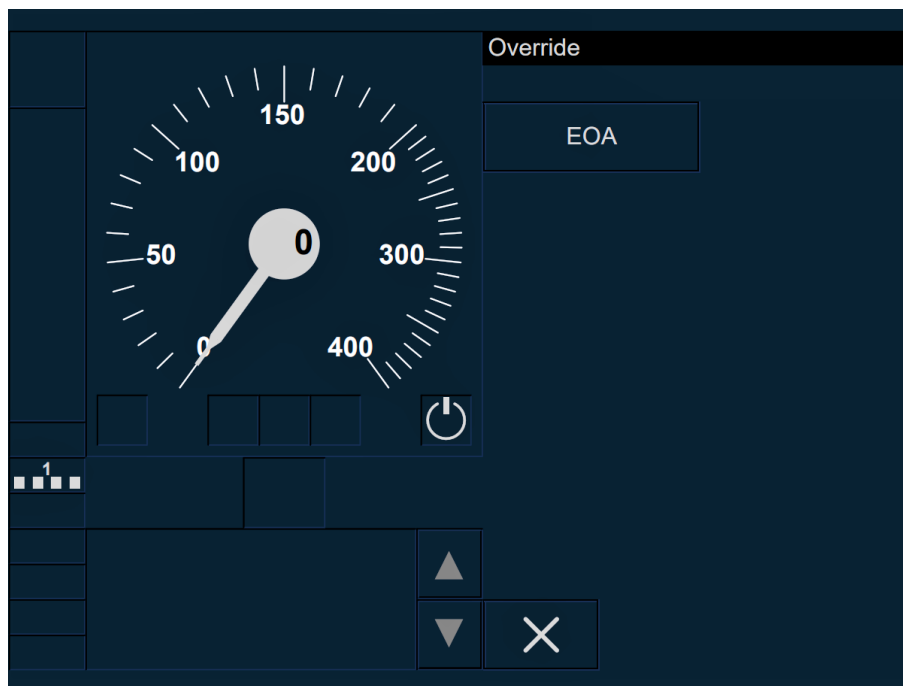


Obrázek 2.19: Hlavní obrazovka, zdroj [1]

### 2.11.1.2 Obrazovka potlačení

Obrazovka potlačení slouží k potlačení konce oprávnění k jízdě. Pro tento účel obsahuje jedno tlačítko s popisem „EOA“, které je možné stlačit pouze za určitých podmínek. K upřesnění chování aplikace po stisknutí daného tlačítka slouží diagram v kapitole 10.7.5 v [1]. Viz kapitola 10.2.2 v [1].





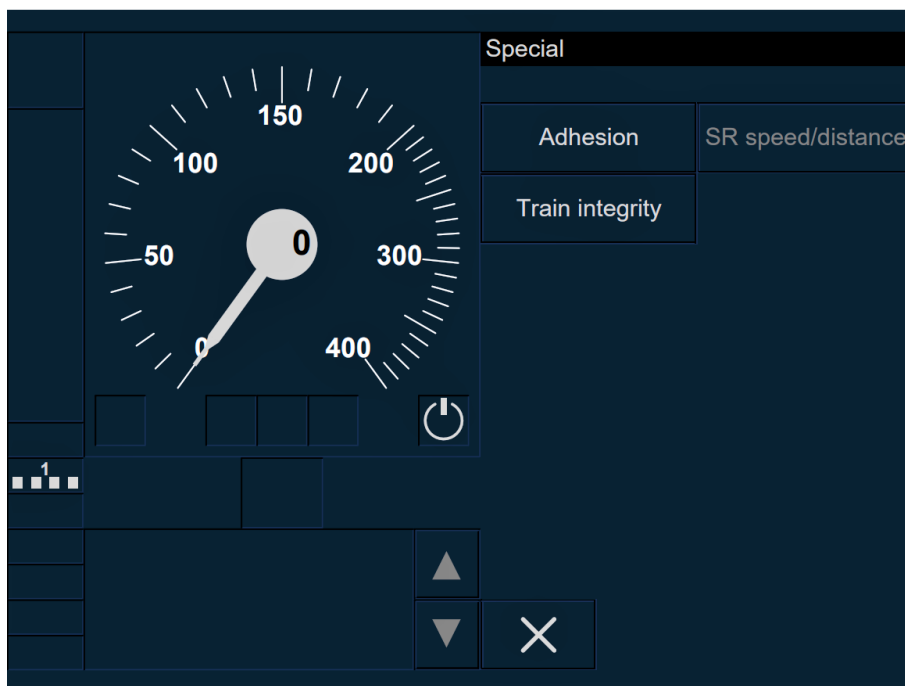
Obrázek 2.20: Obrazovka potlačení, zdroj [1]

### 2.11.1.3 Speciální obrazovka

Účelem speciální obrazovky je zobrazení následujících tří tlačítek, viz kapitola 10.2.3 v [1]:

- Adheze (Adhesion) – Pro přepnutí na obrazovku adheze.
- Rychlost/vzdálenost v SR (SR speed/distance) – Pro přepnutí na obrazovku rychlosti/vzdálenosti v SR.
- Celistvost vlaku (Train integrity) – Pro aktivaci funkce „Celistvost vlaku“.

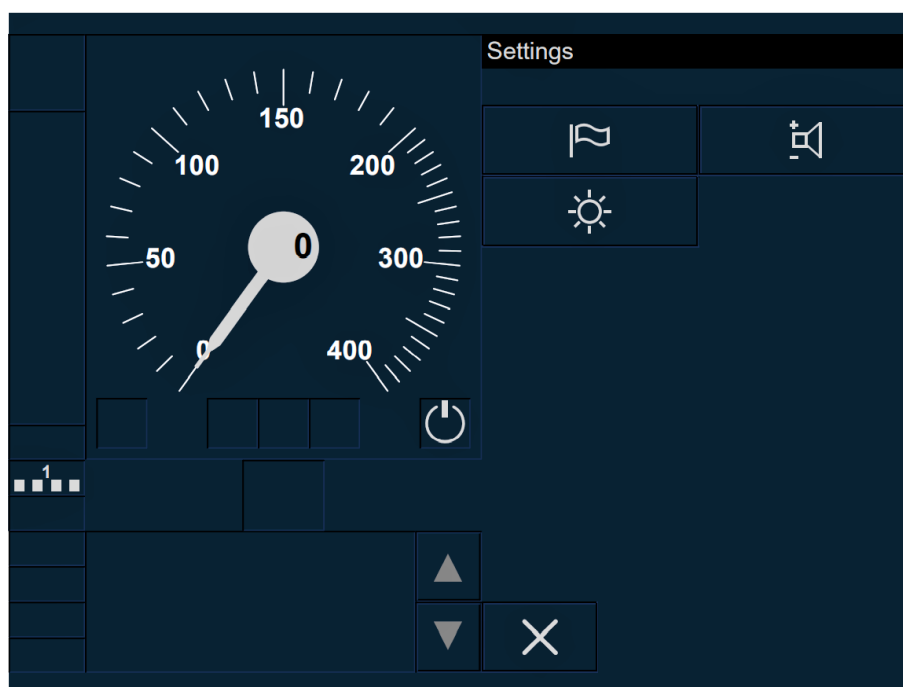
Definice chování aplikace po stisknutí prvních dvou tlačítek, které přepínají obrazovky, zajišťuje diagram v kapitole 10.7.6 v [1].



Obrázek 2.21: Speciální obrazovka, zdroj [1]

### 2.11.1.4 Obrazovka nastavení

Obrazovka nastavení umožňuje strojvedoucímu nastavit si zvolený jazyk, úroveň jasu a hlasitost aplikace. K tomuto úkonu jsou v obrazovce tři tlačítka s příslušnými symboly, které označují vlastnost k přenastavení (jazyk, hlasitost a jas) a přepínají na náležitou obrazovku. Pro názorné zobrazení chování aplikace a obrazovky nastavení je v kapitole 10.7.7 v [1] umístěn diagram. Viz kapitola 10.2.4 v [1].

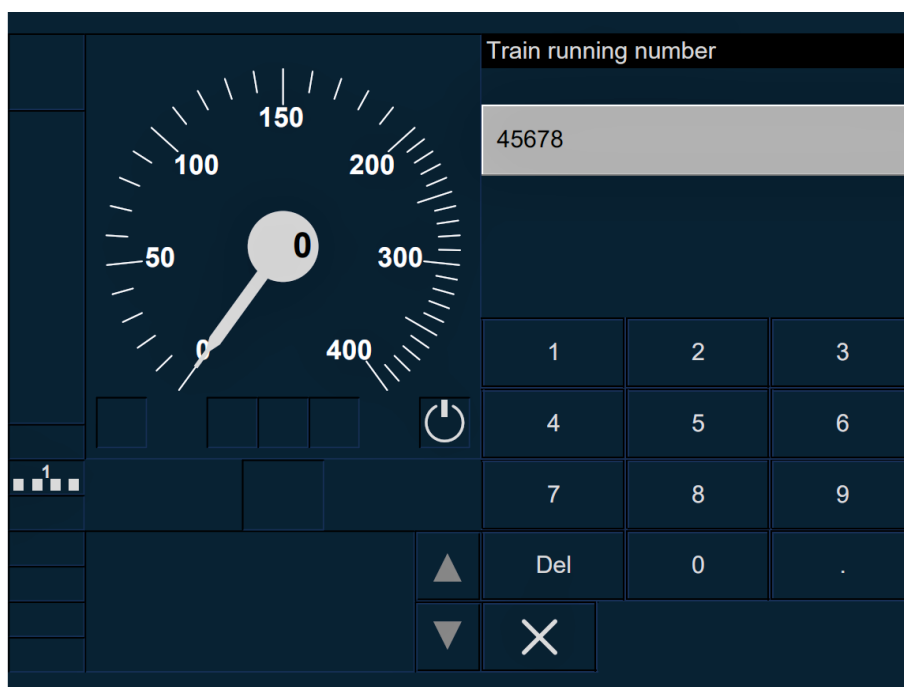


Obrázek 2.22: Obrazovka nastavení, zdroj [1]

## 2.11.2 Obrazovky pro zadávání dat

### 2.11.2.1 Obrazovka čísla vlaku

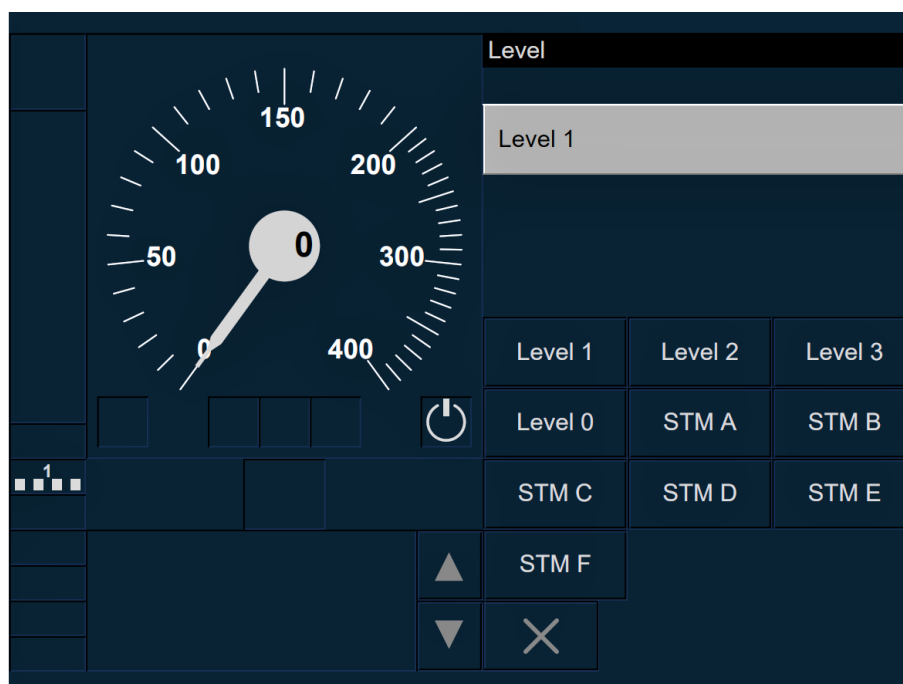
Účelem této obrazovky je zadání/validace čísla vlaku. Z tohoto důvodu obrazovka obsahuje jedno pole pro vstup a numerickou klávesnici. Viz kapitola 10.3.1 v [1].



Obrázek 2.23: Obrazovka čísla vlaku, zdroj [1]

### 2.11.2.2 Obrazovka úrovně

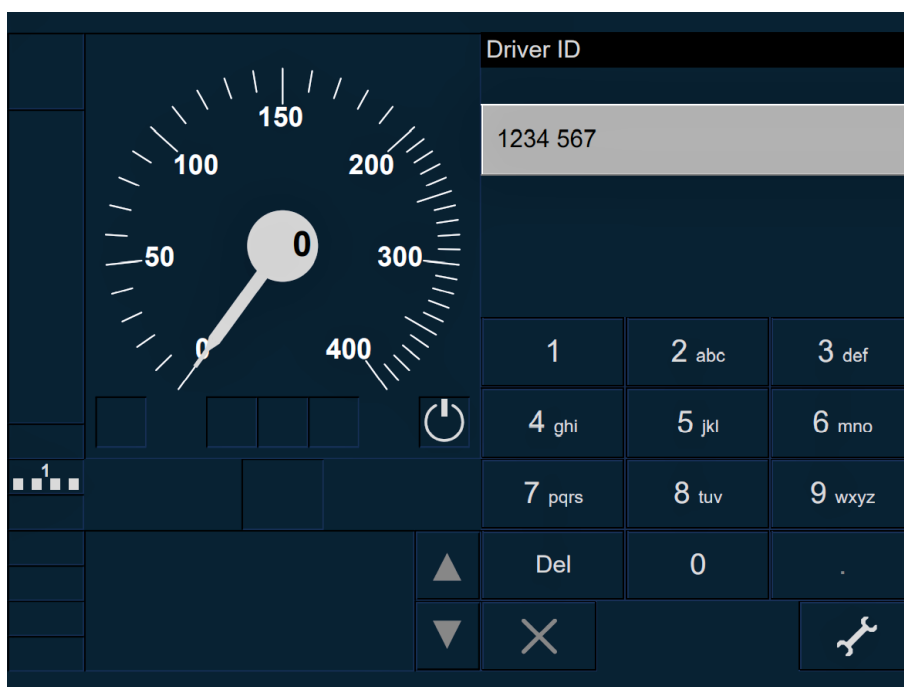
Jedná se o obrazovku, která umožňuje zadávání/validaci ERTMS/ETCS úrovně pomocí jednoho pole pro vstup a dedikované klávesnice, jejíž klávesy reprezentují jednotlivé možnosti ERTMS/ETCS úrovně. Viz kapitola 10.3.2 v [1].



Obrázek 2.24: Obrazovka úrovně, zdroj [1]

### 2.11.2.3 Obrazovka čísla strojvedoucího

Obrazovka slouží pro zadávání/validaci čísla strojvedoucího. K tomuto účelu je přiřazena alfanumerická klávesnice a jedno pole pro vstup. Viz kapitola 10.3.3 v [1].



Obrázek 2.25: Obrazovka čísla strojvedoucího, zdroj [1]

### 2.11.2.4 Obrazovka kontaktních údajů RBC

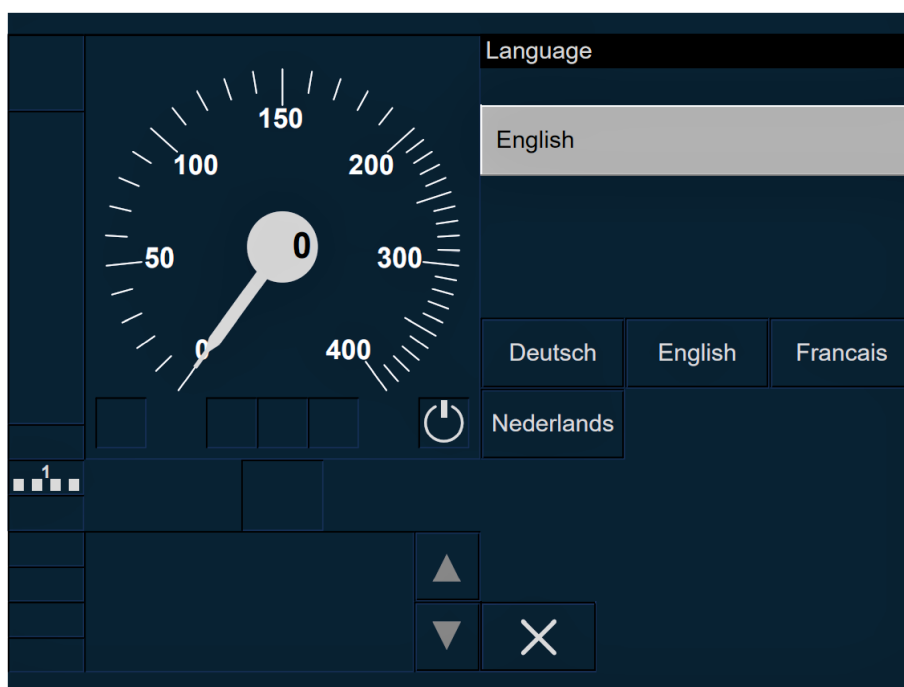
Tato obrazovka slouží pro zadávání/validaci kontaktních údajů modulu RBC. Obsahuje tři pole pro vstup, které pokrývají následující data: číslo rádiové sítě (Radio network ID), číslo RBC (RBC ID), telefonní číslo RBC (RBC telephone number). Všechny tři pole pro vstup mají přiřazené numerické klávesnice a v levé polovině obrazovky se nachází tlačítko pro potvrzení zadaných dat. Viz kapitola 10.3.4 v [1].

RBC contact information		
Radio network ID	1234 56	
RBC ID	1234 5678	
RBC telephone number	1234 5678 9012 3456	
	1	2
	4	5
	7	8
	9	
RBC contact information entry complete?	Del	0
	.	
Yes	✕	

Obrázek 2.26: Obrazovka kontaktních údajů RBC, zdroj [1]

#### 2.11.2.5 Obrazovka jazyku

Účelem obrazovky je zadávání/validace jazyku aplikace. Obsahuje jedno pole pro vstup a dedikovanou klávesnici, jejíž klávesy reprezentují jednotlivé jazyky, které jsou nakonfigurované v systému. Viz kapitola 10.3.5 v [1].



Obrázek 2.27: Obrazovka jazyku, zdroj [1]

### 2.11.2.6 Obrazovka hlasitosti

Obrazovka je určena k zadávání/validaci hlasitosti zvuků aplikace, k čemuž slouží jedno pole pro vstup. Typ klávesnice je poté dle kapitoly 10.3.6 v [1] záležitost, kterou dokumentace přenechává na implementaci. Doporučené možnosti jsou tvořeny například z: několik tlačítek s různými úrovněmi hlasitosti, nebo dvě tlačítka pro zvýšení/snížení hlasitosti, atd. Viz kapitola 10.3.6 v [1].

### 2.11.2.7 Obrazovka jasu

Úkolem obrazovky je umožnění zadávání/validace jasu aplikace, čehož se docílí pomocí jednoho pole pro vstup a klávesnice, která je závislá na implementaci, stejně jako v případě obrazovky hlasitosti. Klávesnice může tedy dle kapitoly 10.3.7 v [1] být složená například z: výběru tlačítek s různými úrovněmi jasu, nebo dvou tlačítek pro zvýšení/snížení úrovně jasu, atd. Viz kapitola 10.3.7 v [1].



### 2.11.2.8 Obrazovka/y vlakových dat

V závislosti na typu vlaku, jsou pro zadávání dat o vlaku, dle kapitoly 10.3.8 v [1], využity dvě nebo jedna obrazovka. Tyto obrazovky se poté rozkládají přes celou zobrazovací plochu aplikace, přičemž v levé polovině obrazovky se nacházejí případné echo texty a tlačítko pro potvrzení ukončení zadávání hodnot. Pravá polovina pak zahrnuje pole pro vstup a příslušné klávesnice pro vkládání dat, popřípadě 2 klávesy na přepínání mezi obrazovkami v případě, že je obrazovek více. Viz kapitola 10.3.8 v [1].

Train data (1/2)		
Train category	combi 1	
Airtight	no	
Brake percentage	135	
Max speed (km/h)	160	
combi 1	combi 2	combi 3
combi 4	combi 5	combi 6
combi 7	combi 8	combi 9
Train data entry complete?		
Yes	X	◀ ▶

Obrázek 2.28: Obrazovka vlakových dat, zdroj [1]

### 2.11.2.9 Obrazovka rychlosti/vzdálenosti v SR

Obrazovka je určená k zadávání/validaci dat ohledně rychlosti v SR (SR speed) a vzdálenosti v SR (SR distance), čehož se docílí pomocí dvou polí pro vstup a numerické klávesnice pro obě tyto pole. Rozkládá se přes celou zobrazovací plochu a v levé polovině obsahuje tlačítko pro potvrzení zadaných dat. Viz kapitola 10.3.9 v [1].

## 2. ANALÝZA DOSTUPNÉ DOKUMENTACE

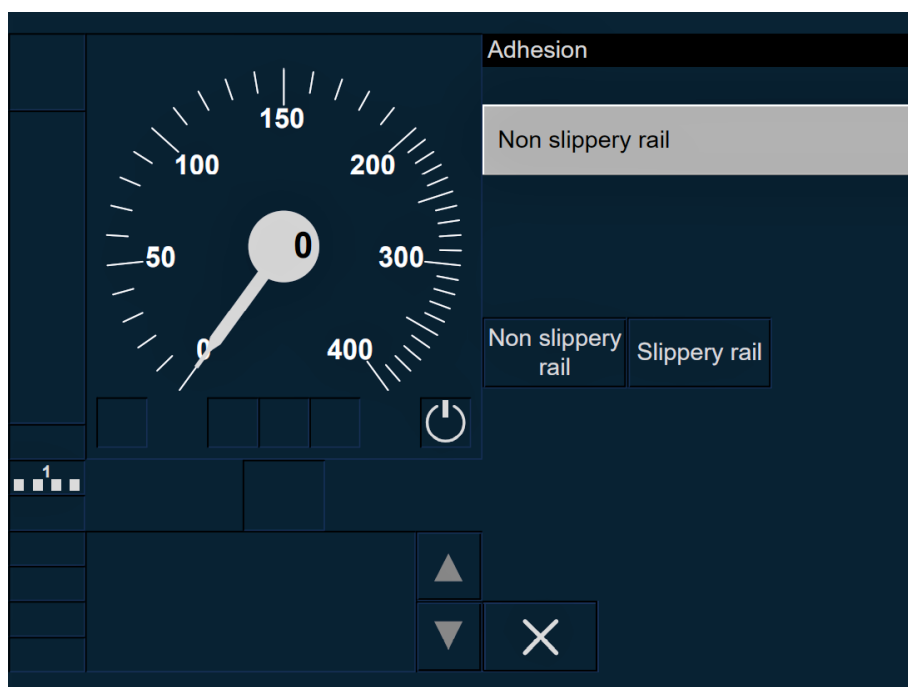
---

SR speed/distance	
SR speed (km/h)	40
SR distance (m)	450
SR speed/distance entry complete?	
Yes	X

Obrázek 2.29: Obrazovka rychlosti/vzdálenosti v SR, zdroj [1]

### 2.11.2.10 Obrazovka adheze

Tato obrazovka, která zabírá pravou polovinu zobrazované plochy aplikace, slouží pro zadávání/validaci adheze. Pro tento účel je obsaženo jedno pole pro vstup a dedikovaná klávesnice, která zahrnuje pouze dvě tlačítka se stavy adheze trati: mokré kolejnice (Slippery rail), suché kolejnice (Non slippery rail). Viz kapitola 10.3.10 v [1].

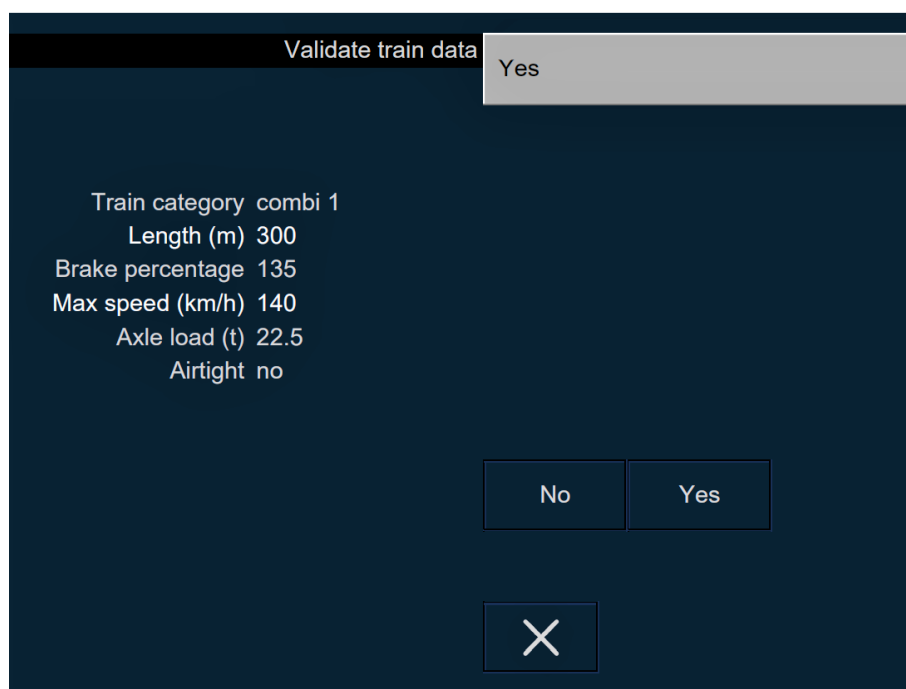


Obrázek 2.30: Obrazovka adheze, zdroj [1]

### 2.11.3 Obrazovky ověření dat

#### 2.11.3.1 Obrazovka ověření vlakových dat

Obrazovka se rozkládá přes celou zobrazovací plochu aplikace a jejím účelem je validace vlakových dat pomocí jednoho pole pro vstup a dedikované klávesnice se dvěma tlačítky: Ano, Ne. Data jsou prezentována v levé polovině obrazovky s využitím echo textů. Viz kapitola 10.4.1 v [1].

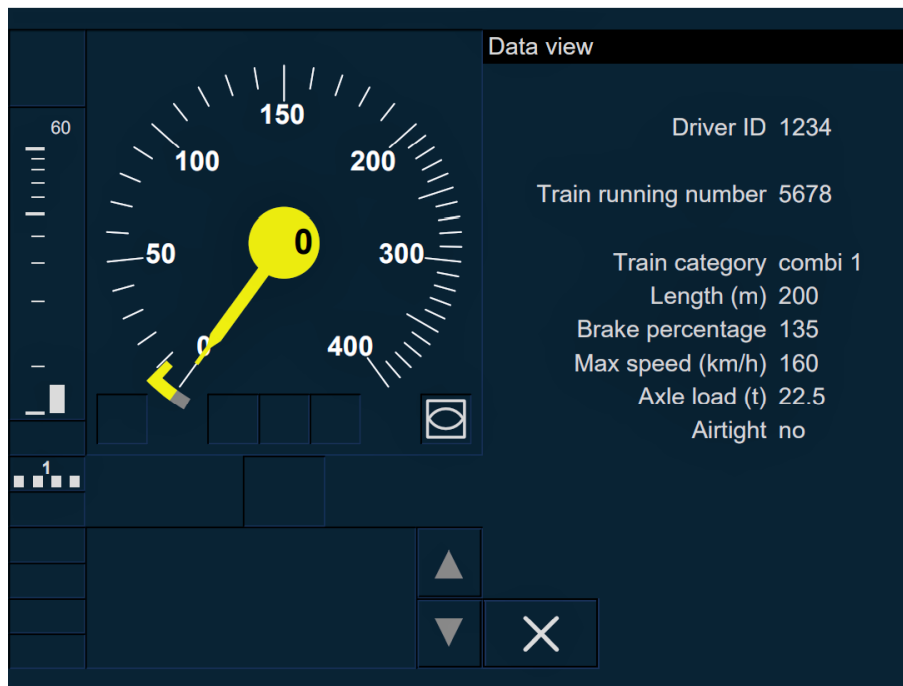


Obrázek 2.31: Obrazovka ověření vlakových dat, zdroj [1]

### 2.11.4 Obrazovky prohlížení dat

#### 2.11.4.1 Obrazovka prohlížení dat

Účelem této obrazovky je zobrazit strojvedoucímu aktuální hodnoty dat, které se zadávají v obrazovce čísla strojvedoucího, obrazovce čísla vlaku a obrazovce vlakových dat. Viz kapitola 10.5.1 v [1].



Obrázek 2.32: Obrazovka prohlížení dat, zdroj [1]

## 2.12 Dialogové sekvence

Dialogové sekvence popisují formou názorných diagramů kroky, které aplikace vykonává při určitých událostech nebo procedurách. Jednotlivé kroky se pak týkají například logiky a podmínek pro přepínání obrazovek či zobrazování určitých symbolů. Přechody mezi stavy diagramů závisí na vstupu, který DMI přijme buď od strojvedoucího nebo od modulu EVC.

### Zahájení mise

Jedná se o proceduru, která se inicializuje po zapnutí aplikace při splnění počátečních podmínek. Hlavním účelem této procedury je validace a potvrzení určitých dat a navázání spojení s modulem RBC přes modul EVC. Dokumentace pomocí diagramu popisuje všechny jednotlivé kroky a chování aplikace. Viz kapitola 10.7.2 v [1].

### Obrazovky prohlížení dat

Diagram popisuje interakci strojvedoucího s aplikací v případě zobrazené hlavní obrazovky, která se vykreslí po stisknutí tlačítka „Hlavní“ ve výchozí obrazovce nebo během procedury zahájení mise. Viz kapitola 10.7.3 v [1].

### **Posun**

Diagram znázorňuje interakci strojvedoucího s aplikací poté, co stiskl tlačítko „Posun“ v hlavní obrazovce. Viz kapitola 10.7.4 v [1].

### **Obrazovka potlačení**

Diagram zobrazuje interakci strojvedoucího s aplikací v případě, že stiskl tlačítko „Potlačení“ ve výchozí obrazovce. Viz kapitola 10.7.5 v [1].

### **Speciální obrazovka**

Diagram znázorňuje způsob interakce strojvedoucího s aplikací po stisknutí tlačítka „Speciální“ ve výchozí obrazovce, přičemž dojde k přepnutí na speciální obrazovku. Viz kapitola 10.7.6 v [1].

### **Obrazovka nastavení**

Diagram zobrazuje interakci strojvedoucího s aplikací v případě, že stiskl tlačítko se symbolem nastavení ve výchozí obrazovce, kdy dojde k přepnutí na obrazovku nastavení. Viz kapitola 10.7.7 v [1].

---

# Analýza diplomové práce Petra Stříteského

V druhé části analýzy se zaměřím na diplomovou práci Petra Stříteského, studenta Fakulty dopravní ČVUT v Praze, která nese název „Generický návrh provozních displejů drážních vozidel“. Jedním z pokynů této práce je návrh a vývoj univerzální grafické knihovny pro zobrazení displejů drážních vozidel. Na této knihovně je pak postavena aplikace DMI, která využívá její funkce pro vykreslování. Z tohoto důvodu se tedy při analýze zaměřím zejména na část zabývající se knihovnou.

## 3.1 Použité technologie

Knihovna je vytvořena pomocí jazyků C a C++ vzhledem k jejich využití napříč různými platformami. Základ, na kterém je knihovna vyvinuta, tvoří grafická knihovna SDL ve verzi 2.0, jejíž funkce jsou zde využité. Zvoleným vývojovým prostředím pro vývoj knihovny je pak Microsoft Visual Studio 2019 ve verzi Professional. Viz kapitola 4 v [2].

## 3.2 Obsah knihovny

### 3.2.1 Soubory

Knihovna zahrnuje následující soubory, viz kapitola 8 v [2]:

- Hlavičkový soubor s definicemi proměnných, které využívají její funkce
- Hlavičkový soubor s definicemi všech funkcí, které knihovna uživateli nabízí
- Soubor s těly funkcí nabízených knihovnou

#### 3.2.2 Funkce

Pro vykreslení objektů, které displeje drážních vozidel vyžadují, obsahuje tato knihovna nejen obecně zaměřené funkce, ale i funkce směřované na specifické zobrazení kolejových vozidel. Jejich účely jsou následující, viz kapitola 7 v [2]:

- Vykreslení geometrických tvarů
- Vykreslení textů
- Vykreslení obrázků
- Vykreslení tlačítek
- Vykreslení charakteristických elementů obrazovek kolejových vozidel

#### 3.3 Doporučené postupy pro použití knihovny

Pro zpřehlednění kódu je v kapitole 8.1 v [2] uvedena doporučená struktura systému souborů ve formě názorného diagramu. Tato struktura je doporučená, ale pro konkrétní potřeby aplikace je samozřejmě nutné si ji případně i přizpůsobit. Dále jsou zde v několika podkapitolách podrobněji popsány stěžejní soubory, které knihovna používá.

Součástí diplomové práce je i návod pro zavedení knihoven SDL 2.0 do projektu ve vývojovém prostředí Microsoft Visual Studio. Viz kapitola 8.2.2 v [2]. Samotnou knihovnu od Petra pak stačí pouze vložit do projektového adresáře a její funkce lze použít pomocí direktivy `#include` v souborech s kódem, ve kterých je budeme využívat.

##### 3.3.1 Problém s knihovnou

Při použití knihovny se objevil pouze jeden problém ovlivňující výkon aplikace. U funkcí, které slouží k vykreslení textů, se při každém jejich volání vytvářejí nové textury, což snižuje výkon. Z tohoto důvodu bylo potřeba knihovnu upravit tak, aby se textury nevytvářely tak často. Úpravy jsou popsány v kapitole 5.3.



---

# Funkční a nefunkční požadavky

Na základě analýzy dokumentace a konzultací se zadavateli z Fakulty dopravní ČVUT jsou definovány následující funkční a nefunkční požadavky pro aplikaci DMI displeje.

## 4.1 Funkční požadavky

### F1 – Komunikace se strojvedoucím

Displej bude předávat strojvedoucímu data prostřednictvím obrazovky. Strojvedoucí ovládá systém stiskem tlačítka na dotykové obrazovce. Aktivace tlačítka bude doprovázena vizuální a zvukovou zpětnou vazbou.

### F2 – Komunikace s modulem EVC

Aplikace bude vyžadovat pro její fungování spojení s modulem EVC, které bude realizováno za pomoci technologie MQTT protokolu. DMI bude ve formě klienta komunikovat s modulem EVC přes společný MQTT server.

### F3 – Rychloměr

Aplikace bude ve výchozí obrazovce zobrazovat kruhový rychloměr s ručičkou, znázorňující aktuální rychlost vlaku. Oblast rychloměru bude obsahovat také rychlostní stupnici, a to v rozmezí rychlostí 0 až 400 km/h.

### F4 – Kruhový ukazatel rychlosti

Aplikace bude ve výchozí obrazovce zobrazovat funkční kruhový ukazatel rychlosti, který bude vyobrazovat příslušné rychlosti v závislosti na stavu dohledu a také na základě brzděné křivky. V souvislosti s touto logikou je součástí i barevné odlišení těchto rychlostí.

##### **F5 – Brzdné informace**

Výchozí obrazovka bude zobrazovat vzdálenost k cíli ve formě pohyblivého obdélníku, znázorňující onu vzdálenost, společně s textovou formou. Dále bude zobrazen indikátor zásahu nouzové/provozní brzdy a také bude umožněno vypnutí/zapnutí zobrazení informací o rychlosti.

##### **F6 – Doplnkové informace**

Ve výchozí obrazovce bude aplikace zobrazovat informace o aktuálním módu, aktuální úrovni a volné koleji před vlakem. Dalšími vykreslovanými elementy budou obdržené textové zprávy, příkazy a oznámení o stavu trati a také indikátor adhezního faktoru.

##### **F7 – Plánovací oblast**

Výchozí obrazovka bude vykreslovat plánovací oblast se stupnicí vzdáleností, mezi nimiž lze přepínat pomocí dvou tlačítek, které reprezentují přiblížení vzdálenosti a oddálení vzdálenosti. Dalšími vykreslovanými objekty budou příkazy a oznámení, profil gradientů, změny rychlostního profilu a také rychlostní profil plánovací oblasti. Dále se v plánovací oblasti bude zobrazovat také indikační značka a tlačítko pro schování/zobrazení.

##### **F8 – Monitorovací informace**

Aplikace bude ve výchozí obrazovce zobrazovat indikaci komunikační relace s modulem RBC, indikaci povolení couvání, lokální čas, a také geografickou pozici vlaku.

##### **F9 – Výběr tlačítek pro přepínání podúrovňových obrazovek**

Ve výchozí obrazovce se bude zobrazovat výběr tlačítek pro přepínání na příslušné obrazovky.

##### **F10 – Hlavní obrazovka**

Aplikace bude zobrazovat hlavní obrazovku, a to jak ve stavu se všemi tlačítky povolenými, tak i se všemi tlačítky vypnutými a pohyblivým symbolem přesýpacích hodin v oblasti názvu této obrazovky.

##### **F11 – Obrazovka potlačení**

Aplikace bude zobrazovat obrazovku potlačení.

**F12 – Speciální obrazovka**

Aplikace bude zobrazovat speciální obrazovku.

**F13 – Obrazovka nastavení**

Aplikace bude zobrazovat obrazovku nastavení.

**F14 – Obrazovka čísla vlaku**

Aplikace bude zobrazovat obrazovku čísla vlaku.

**F15 – Obrazovka úrovně**

Aplikace bude zobrazovat obrazovku úrovně.

**F16 – Obrazovka čísla strojvedoucího**

Aplikace bude zobrazovat obrazovku čísla strojvedoucího.

**F17 – Obrazovka kontaktních údajů RBC**

Aplikace bude zobrazovat obrazovku kontaktních údajů s RBC.

**F18 – Obrazovka jazyku**

Aplikace bude zobrazovat obrazovku jazyku.

**F19 – Obrazovka hlasitosti**

Aplikace bude zobrazovat obrazovku hlasitosti.

**F20 – Obrazovka jasu**

Aplikace bude zobrazovat obrazovku jasu.

**F21 – Obrazovka/y vlakových dat**

Aplikace bude zobrazovat obrazovku/y vlakových dat.

**F22 – Obrazovka rychlosti/vzdálenosti v SR**

Aplikace bude zobrazovat obrazovku rychlosti/vzdálenosti v SR.

**F23 – Obrazovka adheze**

Aplikace bude zobrazovat obrazovku adheze.

### **F24 – Obrazovka ověření vlakových dat**

Aplikace bude zobrazovat obrazovku ověření vlakových dat.

### **F25 – Obrazovka prohlížení dat**

Aplikace bude zobrazovat obrazovku prohlížení dat.

### **F26 – Procedura zahájení mise**

Aplikace bude ihned po zapnutí realizovat proceduru zahájení mise se všemi jejími kroky, které jsou určeny pro modul DMI.

#### **4.1.1 Priorita funkčních požadavků**

Všechny funkční požadavky zmíněné v předchozí kapitole jsou zde rozdělené do tří kategorií v tabulce dle jejich priority.

Požadavky s \* byly již zcela, nebo z části implementovány mnou a kolegy v rámci předmětů BI-SP1, BI-SP2, prázdninové brigády a této bakalářské práce. Přehled rozdělení práce se nachází v kapitole 6.6. Pokud je požadavek aktuálně hotový pouze z části, v závorce pod ním jsou případně vypsány sekce, které chybí. Tabulka reflektuje aktuální stav rozpracovanosti aplikace. Vzhledem k objemu celkového rozsahu všech funkčních požadavků a jejich komplexnosti, byla po dohodě s vedoucím práce naimplementována část požadavků s vysokou prioritou.

Tabulka 4.1: Tabulka priority funkčních požadavků

<b>Vysoká priorita</b>	<p>F1 – Komunikace se strojvedoucím *</p> <p>F2 – Komunikace s modulem EVC *</p> <p>F26 – Procedura zahájení mise * (<i>nekompletní</i>)</p> <p>F3 – Rychloměr *</p> <p>F4 – Kruhový ukazatel rychlosti *</p> <p>F5 – Brzdné informace *</p> <p>(<i>chybí indikátor zásahu nouzové/provozní brzdy</i>)</p> <p>F6 – Doplnkové informace *</p> <p>(<i>chybí informace o volné koleji před vlakem, textové zprávy, příkazy a oznámení o stavu trati, informace o úrovni jsou nekompletní</i>)</p> <p>F7 – Plánovací oblast *</p> <p>(<i>chybí příkazy a oznámení</i>)</p> <p>F8 – Monitorovací informace *</p> <p>(<i>chybí indikace povolení couvání, lokální čas a geografická pozice vlaku</i>)</p> <p>F9 – Výběr tlačítek pro přepínání podúrovňových obrazovek *</p> <p>F10 – Hlavní obrazovka *</p> <p>F16 – Obrazovka čísla strojvedoucího *</p> <p>F15 – Obrazovka úrovně</p> <p>F17 – Obrazovka kontaktních údajů RBC</p> <p>F21 – Obrazovka/y vlakových dat</p> <p>F24 – Obrazovka ověření vlakových dat</p> <p>F13 – Obrazovka nastavení</p> <p>F18 – Obrazovka jazyku</p>
<b>Střední priorita</b>	<p>F25 – Obrazovka prohlížení dat *</p> <p>F14 – Obrazovka čísla vlaku</p> <p>F11 – Obrazovka potlačení</p> <p>F12 – Speciální obrazovka</p> <p>F23 – Obrazovka adheze</p> <p>F22 – Obrazovka rychlosti/vzdálenosti v SR</p>
<b>Nízká priorita</b>	<p>F20 – Obrazovka jasu</p> <p>F19 – Obrazovka hlasitosti</p>

### 4.2 Nefunkční požadavky

#### **N1 – Programovací jazyk a operační systém**

Aplikace bude implementována v jazyce C++, a to pro operační systém Windows, Linux Debian 11.3.0 a Linux Debian 6.0.10, který je umístěn v průmyslovém displeji.

#### **N2 – Grafická knihovna**

Pro vykreslování grafických částí se bude využívat grafická knihovna od Petra Stříteského, studenta Fakulty dopravní ČVUT v Praze. Knihovna je psána v jazyce C, C++ a využívá funkcí grafické knihovny SDL 2.0.

#### **N3 – Ovládání obrazovky**

Aplikace se bude ovládat pomocí dotykové obrazovky.

#### **N4 – Jazyk**

Aplikace bude podporovat anglický i český jazyk.

#### **N5 – Vzhled a chování**

Vzhled a chování aplikace odpovídá dokumentu ERA-ERTMS-015560v2.3.

#### **N6 – Rozlišení**

Rozlišení aplikace bude 800 x 600 pixelů.

#### **N7 – Latence**

Doba odezvy aplikace bude 100 milisekund.

---

# Návrh

Tato kapitola se zabývá návrhem aplikace a její aktuální architekturou. Zde popsaná verze návrhu vychází z původní podoby návrhu aplikace, který byl vytvořen s kolegy z předmětu BI-SP1 v rámci jeho náplně, viz [3].

Později došlo k určitým změnám, popsaným v poslední podkapitole této části, viz 5.9, na základě kterých byl návrh upraven, a jeho aktuální podobu prezentuje právě tato kapitola práce. Grafická podoba aplikace a její chování se zde nenacházejí, protože jsou založené čistě na pokynech v technické dokumentaci, viz [1].

## 5.1 Komunikace s modulem EVC

Komunikace s modulem EVC je klíčovou součástí aplikace, jelikož vyjma informací, které se zadávají v obrazovkách, slouží pro zobrazování objektů data právě od modulu EVC. Pro tento účel je využit komunikační protokol MQTT, založený na principu „publish/subscribe“, tedy zápis a odběr konkrétních témat. Viz [4].

Klienti se připojují k serveru, tzv. „broker“ a odebírají na něm data z témat, které potřebují, přičemž mají možnost na server také data ke konkrétním tématům posílat a tedy je upravovat. Tento protokol pro spojení klientů a serveru využívá síťového komunikačního protokolu TCP/IP. Viz [4].

V aplikaci představuje odebírané téma modul EVC, se kterým si DMI vyměňuje data. Komunikace probíhá obousměrně, tedy jak z EVC do DMI, tak z DMI do EVC. Aplikace odebírá domluvená data na MQTT brokeru každých 100 milisekund a po jejich přijetí ve formátu JSON je zpracuje. Pro odesílání dat se tato data nejprve překládají do formátu JSON a poté posílají do brokera.

V aplikaci je pro klientskou komunikační část komponenta, která odebírá aktuální hodnoty komunikačních dat z klienta a poskytuje je do jednotlivých oblastí aplikace. Prozatím byla po domluvě s kolegy z týmu EVC zvolena důvěra v předávaná data, tedy se předpokládá korektnost obdržených hodnot.

### 5.2 Zpracování formátu JSON

Dokumenty ve formátu JSON jsou v aplikaci využity jako konfigurační soubory s daty pro vykreslované objekty, ale také pro odesílání dat mezi moduly DMI a EVC. Pro zpracování tohoto formátu napříč aplikací je využita knihovna JSON for Modern C++. Viz [5].

### 5.3 Knihovna Petra Stríteského

U funkcí určených pro vykreslení textů bylo z důvodu optimalizace prototypu nutné upravit hlavičky a jejich těla. V původních podobách těchto funkcí dochází při každém jejich volání k vytváření nových textur, což je náročné na procesor. Vhodnou náhradou je vytvoření potřebných textur na začátku spuštění aplikace a poté už pouze v případě, že je potřeba obsah těchto textur změnit (například když došlo ke změně hodnoty vykreslovaného textu).

Funkce byly upraveny takovým způsobem, aby bylo možné jim pro vykreslení textů předávat potřebné textury jako argument.

### 5.4 Architektura obrazovek

Všechny obrazovky v sobě udržuje hlavní koordinační komponenta. Jejím účelem je zajišťovat přepínání a zobrazování těchto jednotlivých obrazovek. Z tohoto důvodu je využit návrhový vzor Observer, kdy koordinační komponenta poslouchá na aktuálně zobrazených obrazovkách a v případě nutnosti přepnutí pak daná obrazovka upozorní posluchače (což je koordinační komponenta), který zareaguje na signál přepnutím na požadovanou obrazovku. Pro oddělení datového modelu, uživatelského rozhraní a řídicí logiky v jednotlivých obrazovkách, popř. podoblastech obrazovky je využitý návrhový vzor Model-View-Controller (MVC).

*Controller* ve spravované oblasti zajišťuje reakci na události a také aktualizaci potřebných dat uložených v Modelu na základě nových hodnot, které aplikace obdržela od modulu EVC. Zároveň může sám data do modulu EVC v případě potřeby přes klienta odesílat. Zpracovávané události se týkají vstupu uživatele, tedy zaznamenané kliknutí na tlačítko v oblasti, nebo reakce na nová příchozí data.

*Model* pro spravovanou oblast poskytuje zdroj všech potřebných dat, které daná oblast využívá. Značná část dat, které Model využívá, se načítá z konfiguračních souborů ve formátu JSON, které obsahují předdefinované hodnoty, jako například pozice objektů, jejich rozměry nebo barvy.

Komponenta *View* pak slouží k zobrazení obsahu oblasti na základě aktuálních hodnot dat poskytnutých od Modelu oblasti.



### **Výchozí obrazovka**

Výchozí obrazovka je na základě poznatků z analýzy rozdělena do 9 separátních oblastí s objekty k zobrazení. Z tohoto důvodu výchozí obrazovka obsahuje vlastní Model, View a Controller pro každou tuto oblast, které fungují nezávisle na sobě a spravují zobrazování i události týkající se pouze jedné své oblasti.

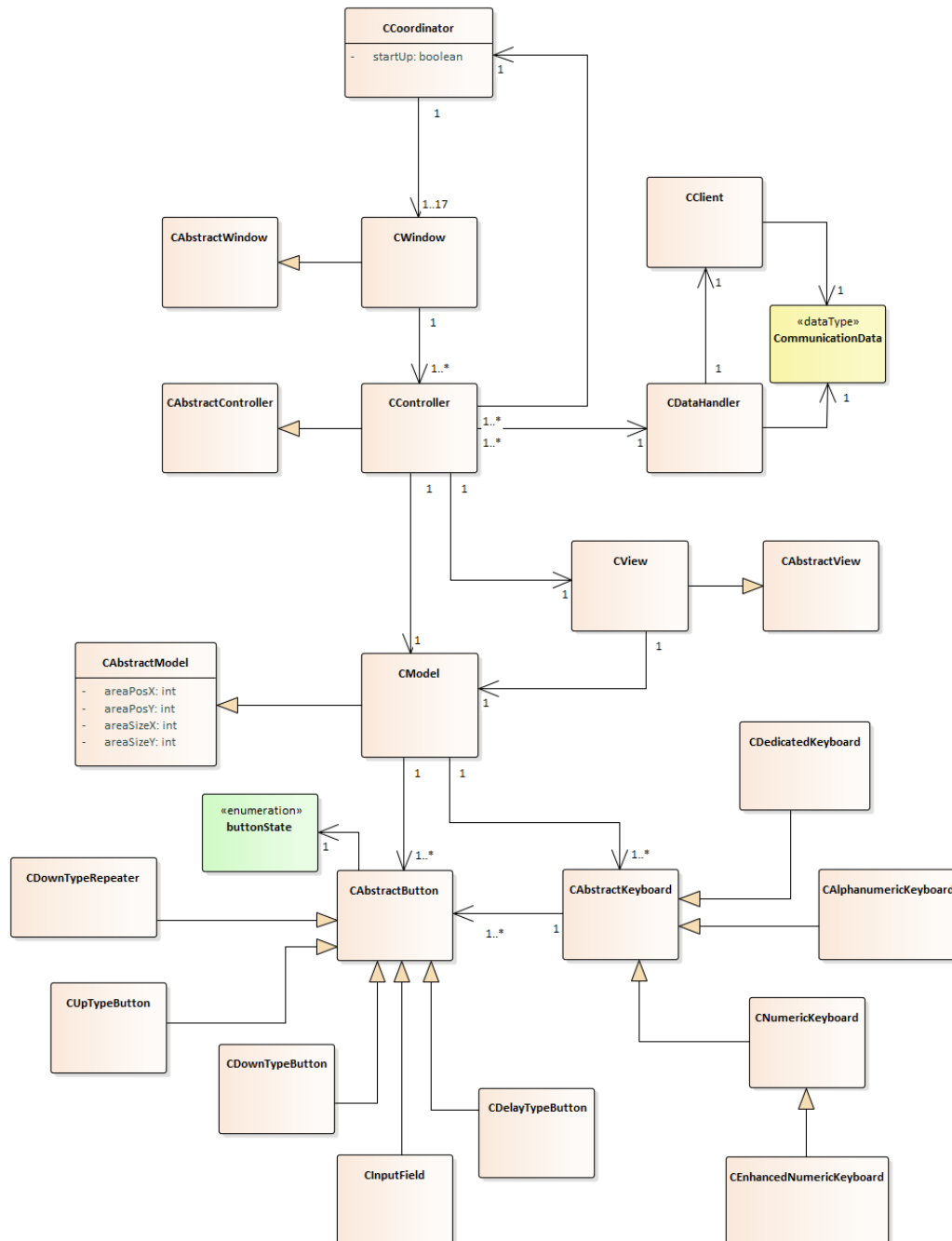
Díky tomuto systému je pak práce s jednotlivými oblastmi přehlednější a jednodušší, vzhledem k obsáhlejší logickým úkonům těchto celků. Pro případ nutnosti propojení dvou oblastí a komunikace mezi jejich Controllery je použit návrhový vzor Observer, při kterém jeden z Controllerů poslouchá na druhém Controlleru a v případě potřeby může svá data upravit na základě upozornění.

### **Ostatní obrazovky**

Ostatní obrazovky vzhledem k menšímu rozsahu jejich logických úkonů obsahují ve většině případů pouze jeden Model, View a Controller, které spravují celou obrazovku.

## 5.5 Doménový model

V této sekci je znázorněn doménový model, který vyjadřuje vztahy a propojení mezi jednotlivými třídami v aplikaci.



Obrázek 5.1: Doménový model

## 5.6 Důležité třídy

Zde jsou přiblíženy důležité třídy, které jsou zobrazeny v doménovém modelu.

### **CCoordinator**

Třída `CCoordinator` reprezentuje koordinační komponentu, která zajišťuje přepínání mezi obrazovkami. Stará se také o chod aplikace.

### **CClient**

Třída `CClient` zajišťuje spojení mezi modulem `DMI` a modulem `EVC`. Jejím hlavním úkolem je odebírat data z MQTT brokeru, která jsou nezbytná pro fungování aplikace, a také potřebná data z `DMI` na broker odesílat, kde si je odebírá právě modul `EVC`.

### **CDataHandler**

Tato třída představuje prostředníka mezi klientem a `Controllery` jednotlivých oblastí, které využívají data obdržaná od klienta. Důvodem pro existenci této třídy je oddělení manipulace s obdržеныmi daty od komunikace s modulem `EVC`.

### **CAbstractController**

Tato třída slouží jako abstraktní šablona pro `Controllery` v aplikaci.

### **CAbstractModel**

Tato třída představuje abstraktní šablonu pro komponenty typu `Model` vyskytující se v aplikaci.

### **CAbstractView**

Třída `CAbstractView` reprezentuje abstraktní šablonu pro komponenty typu `View` v aplikaci.

### **CAbstractWindow**

Třída `CAbstractWindow` představuje abstraktní šablonu pro obrazovky, které se v aplikaci vyskytují.

### **CAbstractButton**

Tato třída reprezentuje šablonu pro všechna tlačítka v aplikaci. Obsahuje také několik funkcí pro vykreslení tlačítek dle konkrétní potřeby vzhledu.

### **CAbstractKeyboard**

Tato třída slouží jako abstraktní šablona pro klávesnice. Svůj výstup, tedy stisknuté tlačítko, předává příslušnému poli pro vstup.

## 5.7 Procesy

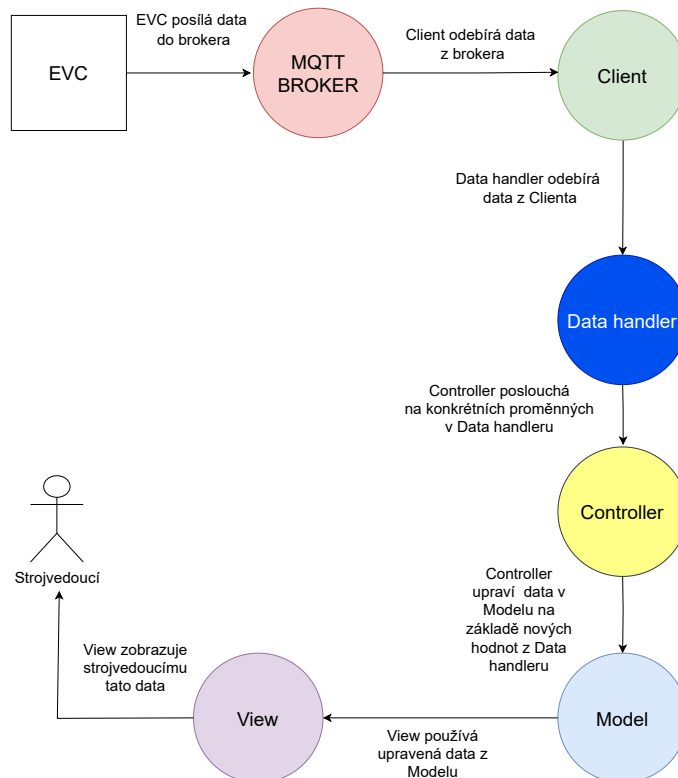
V této části práce přiblížím a popíšu nejdůležitější procesy, které se v aplikaci vykonávají. Ke každému z těchto procesů je také přiložen názorný diagram, který zjednodušuje pochopení dané logiky.

### 5.7.1 Datový tok z modulu EVC do DMI

Následující diagram zachycuje proces putování nových dat od modulu EVC až po jejich zobrazení v aplikaci. Každá oblast spravovaná komponentami MVC takto odebírá potřebná data a následně je zobrazí na obrazovce ve formě příslušných objektů nebo jejich částí.

Proces začíná odesláním dat od modulu EVC na MQTT broker, ze kterého si tato data odebírá komponenta Client v aplikaci. Poté jsou data předána do komponenty Data handler. Následně každý Controller, který poslouchá na určitých potřebných datech u Data handleru pak v případě, že obsahují nové hodnoty, upraví interně data ve svém Modelu.

Komponenty View pak pro zobrazení své oblasti a příslušných objektů využívají nových hodnot, které jsou obsažené v Modelech.

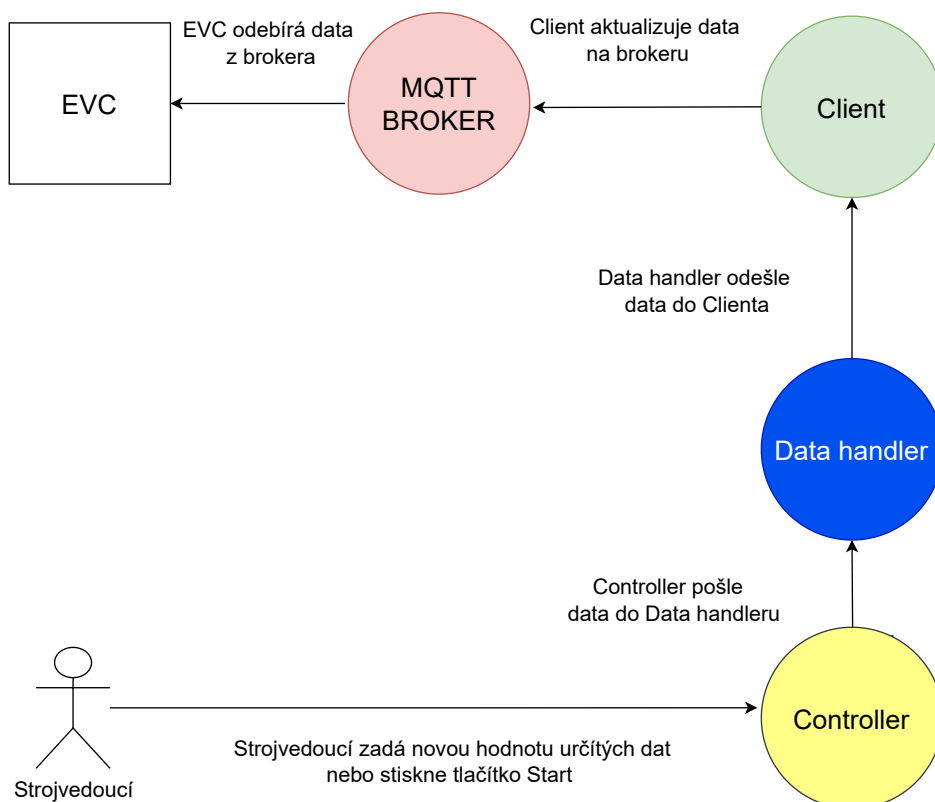


Obrázek 5.2: Komunikace z EVC do DMI

### 5.7.2 Datový tok z DMI do modulu EVC

Následující diagram zachycuje proces odeslání dat do modulu EVC. Proces může začít dvěma způsoby. První způsob je přes zadání nových hodnot dat v některé z obrazovek pro zadávání dat a jejich potvrzení. Druhý způsob je pomocí stisknutí tlačítka Start. Následně Controller dané obrazovky předá tyto data do Data handleru.

Data handler reaguje odesláním dat do Klienta, který provede jejich aktualizaci na brokeru. Z brokeru si poté informace odebere modul EVC.



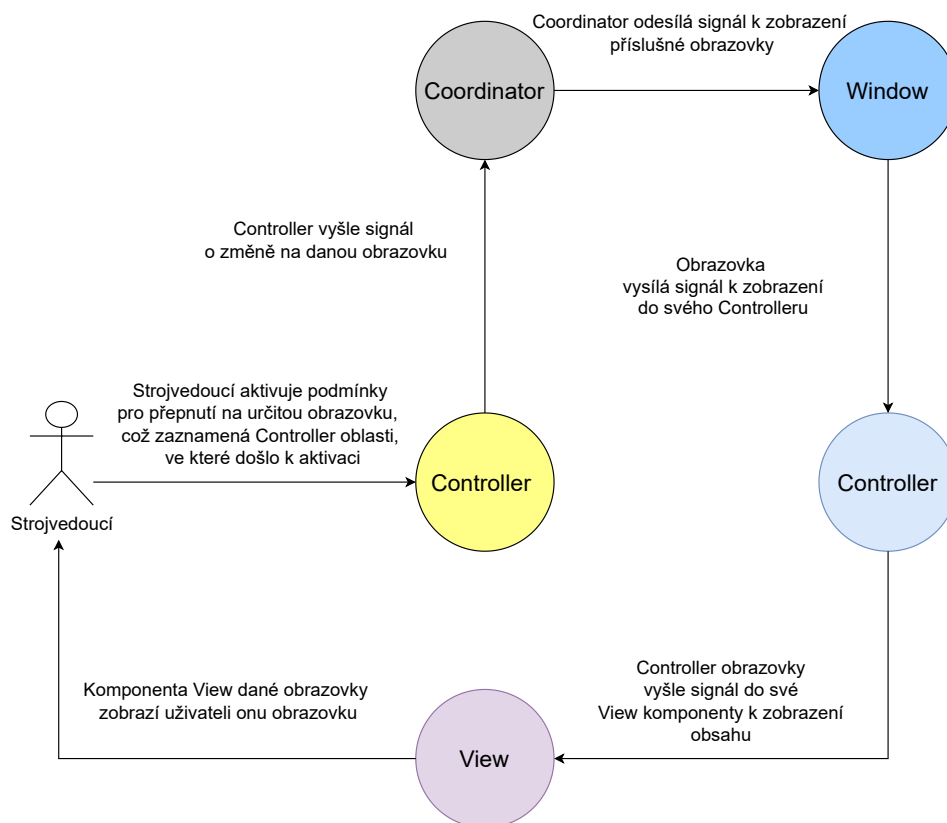
Obrázek 5.3: Komunikace z DMI do EVC

### 5.7.3 Přepnutí obrazovky

Tento diagram reprezentuje proces přepínání z jedné obrazovky na jinou. Přepnutí obrazovky může v závislosti na velikosti požadované obrazovky znamenat zobrazení nové obrazovky přes celou zobrazovací plochu, nebo zobrazení nové obrazovky v pravé polovině zobrazovací plochy a zároveň zobrazení levé poloviny výchozí obrazovky.

Proces začíná splněním podmínek pro přechod na požadovanou obrazovku. Tento impuls zachytí Controller oblasti, ve které došlo ke splnění těchto podmínek. Tato oblast může být jak podoblast ve výchozí obrazovce, tak celá podúrovňová obrazovka. Controller následně upozorní komponentu Coordinator signálem o změně na danou obrazovku. Coordinator poté odesílá signál do dané obrazovky o jejím zobrazení.

Tato obrazovka následně upozorní na zobrazení svůj Controller, nebo Controllers, pokud jich je více než jeden, a ten v návaznosti upozorní svou View komponentu. View komponenta poté zobrazí uživateli onu obrazovku, nebo v případě, že je těchto komponent více, pak zobrazí každá z nich svou oblast, které dohromady kompletují danou obrazovku.



Obrázek 5.4: Přepnutí obrazovek

## 5.8 Přejchodová mapa

V této sekci je pro lepší orientaci provázání jednotlivých obrazovek znázorněné schéma přechodové mapy pro přepínání mezi všemi obrazovkami v aplikaci. Jednotlivé typy přechodů jsou znázorněny šipkami s příslušnými barvami dle způsobu přechodu.

Šipka v černé barvě znamená přepnutí pouze pomocí jednoho tlačítka, tedy konkrétního tlačítka pro danou obrazovku, nebo tlačítka zpět.

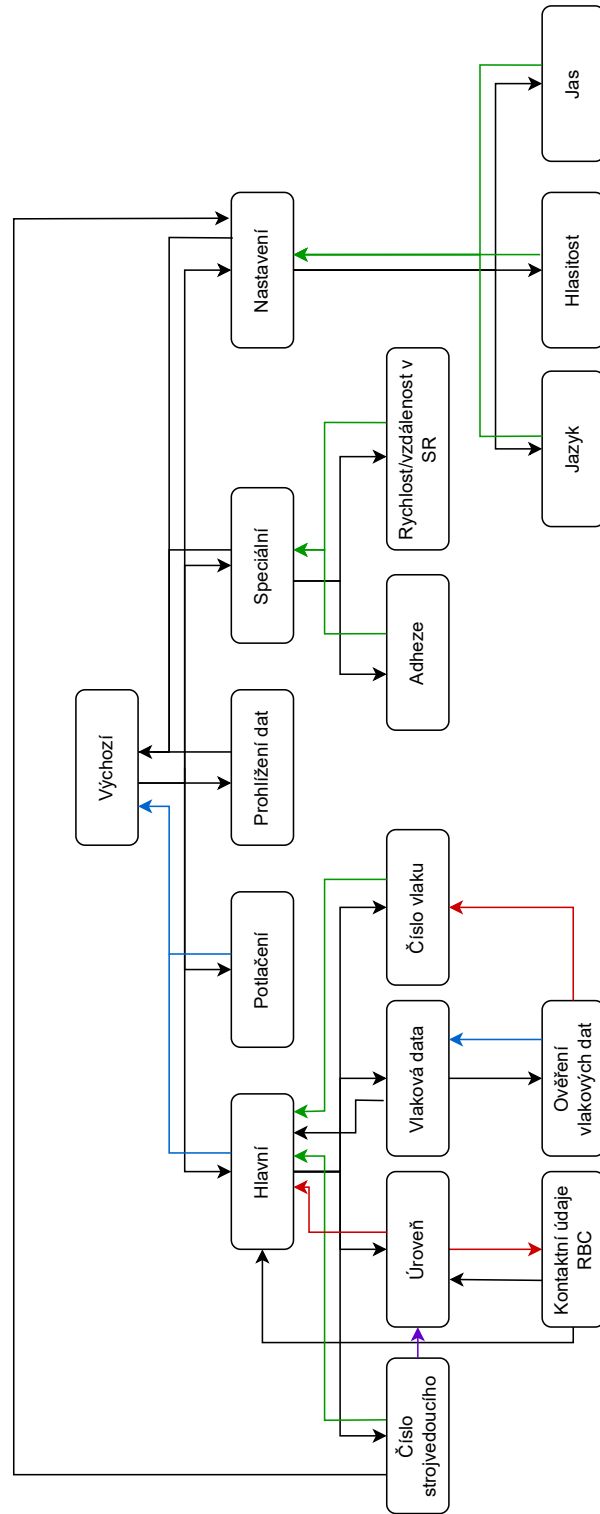
Modrá šipka znázorňuje přepnutí, které je umožněno stisknutím různých tlačítek.

Šipka ve fialové barvě reprezentuje přepnutí na základě specifických podmínek.

Zelená šipka znamená přepnutí ukončením procesu zadávání/validace dat.

Červená šipka reprezentuje přepnutí na základě určité hodnoty zadané do vstupního pole.





Obrázek 5.5: Přechodová mapa obrazovek

## 5.9 Změny oproti původnímu návrhu

Kromě strukturální změny jednotlivých tříd, kdy došlo k upravení určitých funkcí, je hlavní změna v zachytávání uživatelského vstupu. V původní verzi návrhu docházelo k odchyťování a reakcím na uživatelský vstup v komponentách View. Tato logika se pozměnila a přemístila z komponenty View do komponenty Controller.

Při komunikaci s modulem EVC se v původním návrhu vyskytoval také objekt pro validaci dat, které DMI od modulu EVC obdrží. Tento objekt byl nakonec odstraněn a v komunikaci se pouze odchyťávají chyby při zpracování obdržených dat ve formátu JSON, které se vypisují do souboru pro logování těchto chyb.

---

# Implementace

V této kapitole se nacházejí příručky, které poskytují informace k instalaci, vývoji a používání aplikace. Je zde umístěna i ukázka uživatelského rozhraní a schéma projektu vlakového simulátoru. Příručky jsou orientované pouze na operační systémy Windows a Linux Debian 11.3.0, přičemž na jiných operačních systémech se dané pokyny mohou lišit. Průmyslový displej s operačním systémem Linux Debian 6.0.10 bohužel nebyl zprovozněn pro nasazení aplikace a z tohoto důvodu je vynechán.

## 6.1 Instalační příručka

Tento manuál uživateli poskytuje potřebné informace k instalaci a zprovoznění aplikace.

### 6.1.1 Windows

Pro operační systém Windows stačí ke správnému chodu aplikace mít pouze nainstalované knihovny Visual C++ Redistributable od Microsoftu, které lze stáhnout ze stránky: <https://docs.microsoft.com/en-us/cpp/windows/latest-supported-vc-redist?view=msvc-170>. Viz [6].

Poté již stačí si z přiloženého média zkopírovat obsah adresáře *exe* do zvolené složky ve svém počítači a spustit soubor *DMI.exe*.

### 6.1.2 Linux

#### SDL knihovny

Nejprve je nutné stáhnout a nainstalovat knihovny SDL 2.0. Vyhledáme aktuální verze knihoven a poté je nainstalujeme pomocí příkazů, viz [7]:

```
apt-cache search libsdl2
apt-get install libsdl2-dev
```

Dále je potřeba následujícími příkazy nainstalovat knihovny pro práci s fonty, zvuky a obrázky:

```
apt-get install libSDL2-ttf-dev
apt-get install libSDL2-mixer-dev
apt-get install libSDL2-image-dev
```

### Mosquitto knihovna

Knihovnu nainstalujeme pomocí následujícího příkazu:

```
sudo apt install libmosquitto-dev
```

### Spuštění

Zkopírujeme si obsah adresáře *src/impl* z příloženého média do zvolené složky ve svém počítači, přesuneme se do ní pomocí příkazu *cd* a sestavíme aplikaci příkazem *make*. Tím vznikne nový adresář *build*, do kterého přejdeme příkazem *cd* a zde spustíme aplikaci příkazem *./dmi*.

## 6.2 Vývojářská příručka

Tento manuál slouží pro další osoby, které se budou podílet na vývoji této aplikace nebo se budou starat o její údržbu. Obsahuje informace ohledně zprovoznění projektu ve vývojovém prostředí a zavedení potřebných knihoven.

Prvním krokem je instalace knihoven z instalační příručky v kapitole 6.1, až poté se lze věnovat instrukcím v této příručce.

### 6.2.1 Windows

Pro operační systém Windows je doporučené vývojové prostředí Microsoft Visual Studio, ve kterém je aplikace implementována a z toho důvodu bude příručka zaměřena na toto prostředí.

### Git

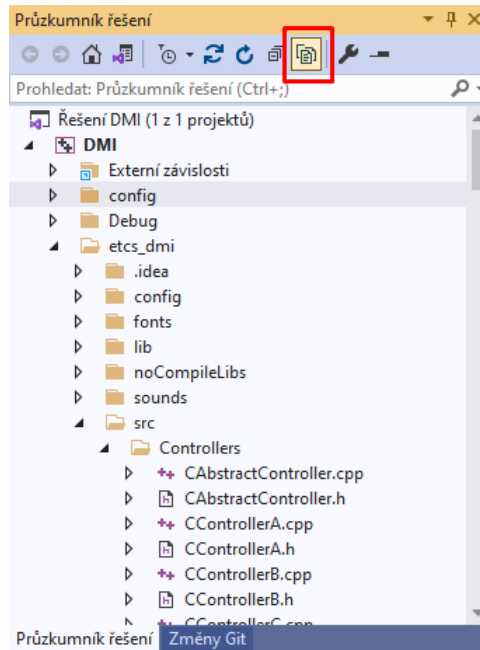
Nejprve naklonujeme větev *master*, pomocí příkazu *git clone* z projektového repozitáře v gitlabu do vybrané složky v počítači.

### Založení projektu ve Visual Studiu

Do stejné složky, ve které je umístěný obsah naklonované větve projektu, si ve Visual Studiu vytvoříme nový prázdný C++ projekt. V této složce nám nyní vznikne nová složka s řešením projektu. Visual Studio vypneme. Obsah této nově vzniklé složky vyjmeme a vložíme jej do složky se soubory naklonovaného projektu, přičemž složku s řešením můžeme nyní odstranit. Visual Studio znovu zapneme. Pro otevření projektu pak zvolíme možnost „Otevřít

projekt nebo řešení“ a zvolíme soubor s příponou `.sln` ve složce s naklonovaným projektem.

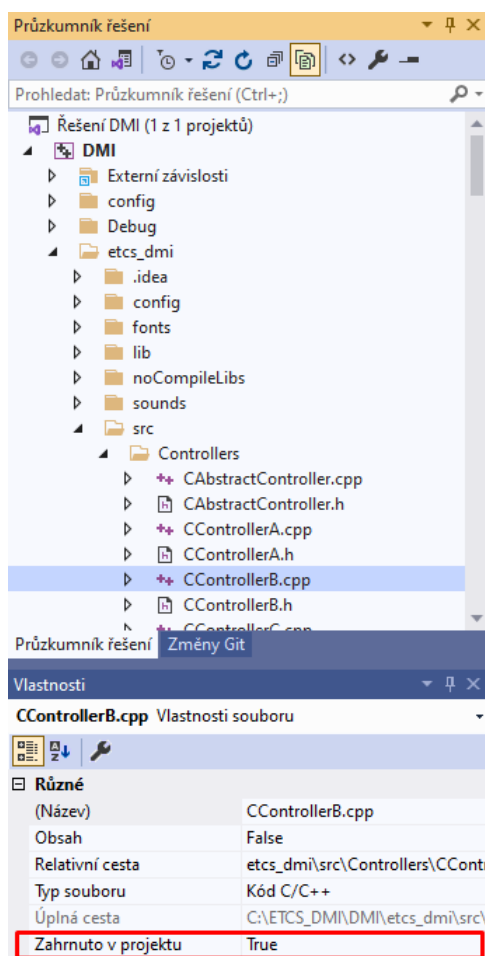
Následně si zvolíme v průzkumníkovi řešení, což je karta, která umožňuje uživateli prozkoumávat soubory a složky řešení, možnost „Zobrazit všechny soubory“ pro zobrazení souborové struktury projektu.



Obrázek 6.1: Průzkumník řešení

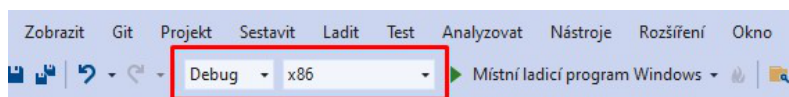
Poté je potřeba pomocí průzkumníka řešení všechny soubory s příponami `.cpp` a `.h` z projektových složek `src` a `lib` zahrnout do projektu. Označíme je a na kartě *Vlastnosti* u těchto souborů změníme hodnotu položky *Zahrnuto v projektu* z `false` na `true`, čímž se nám označené soubory zahrnou do projektu.

## 6. IMPLEMENTACE



Obrázek 6.2: Karta vlastností souboru

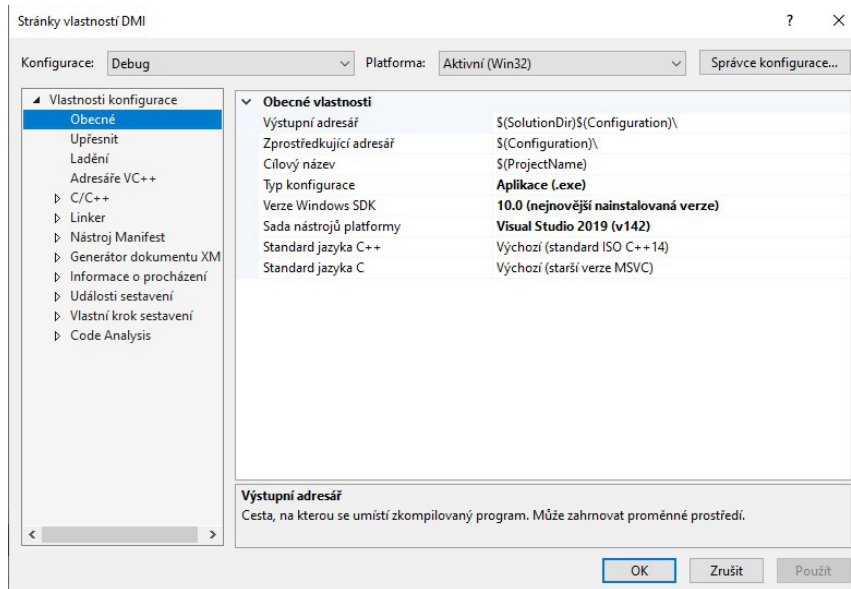
Dále si zvolíme konfiguraci projektu. V horní liště vybereme konfiguraci *Debug* pro platformu *x86*. Následující úpravy importu knihoven jsou vázané na konkrétní konfiguraci, tedy pokud budeme chtít projekt sestavit v jiné konfiguraci, popř. na jinou platformu, musíme tyto úpravy zadat i do nové konfigurace.



Obrázek 6.3: Konfigurace projektu

## Import knihoven

V dalším kroku otevřeme ve Visual Studiu záložku *Projekt* a zde si zvolíme položku *Vlastnosti*.



Obrázek 6.4: Okno vlastností projektu

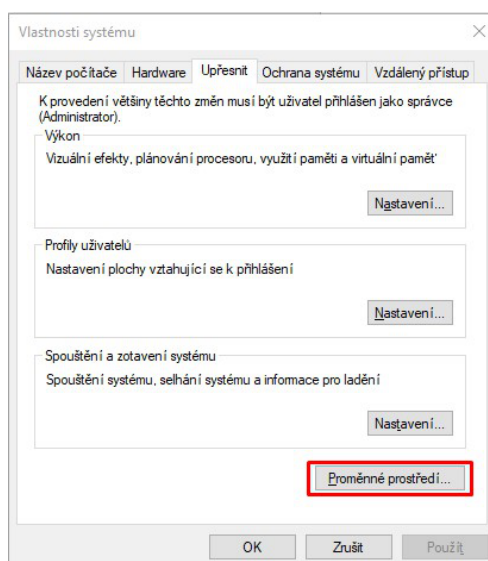
Následně zvolíme kartu s názvem *Adresáře VC++*, kde poté v položce *Adresáře souborů k zahrnutí* přidáme absolutní cestu v souborovém systému k projektové složce `noCompileLibs\SDL2-2.0.14\include`. Dále na kartě *Adresáře VC++* v položce *Adresáře knihoven* vložíme absolutní cestu v souborovém systému k projektové složce `noCompileLibs\SDL2-2.0.14\lib\x86`. Viz [8].

Poté ve stejném okně s vlastnostmi projektu vybereme část *Linker*, ve které pak zvolíme kartu *Vstup* a položku *Další závislosti*. Otevře se dialogové okno, ve kterém vyplníme názvy souborů s příponou `.lib` ze složky vložené v předchozím kroku, jmenovitě tedy: `SDL2.lib`, `SDL2main.lib`, `SDL2_ttf.lib`, `SDL2_image.lib`, `SDL2_mixer.lib`, `mosquitto.lib`. V části *Linker* si poté vybereme kartu *Obecné* a do položky *Další adresáře knihoven* přidáme absolutní cestu v souborovém systému k projektové složce `noCompileLibs\Mosquitto`. Viz [8].

Předposledním krokem je upravení proměnných prostředí operačního systému Windows. Otevřeme okno *Nastavení* a zde vyhledáme položku *Upravit proměnné prostředí systému*. V nově otevřeném okně klikneme na tlačítko s popisem „Proměnné prostředí...“ a rozklikneme proměnnou *Path*, do které přidáme absolutní cestu v souborovém systému k projektové složce `noCompileLibs\SDL2-2.0.14\lib\x86`. Viz [8].

## 6. IMPLEMENTACE

---



Obrázek 6.5: Úprava proměnných prostředí

Nakonec zbývá pouze restartovat Visual Studio, aby používalo aktualizované proměnné prostředí systému.

### 6.2.2 Linux

Aplikace byla vyvíjena na operačním systému Windows, ve vývojovém prostředí Visual Studio, tedy nemám žádné konkrétní doporučení pro operační systém Linux Debian 11.3.0. Z tohoto důvodu přenechám výběr na vývojáři.

### Git

Do vybrané složky si naklonujeme větev *master* pomocí příkazu `git clone` z projektového repozitáře v gitlabu.



## 6.3 Programátorská příručka

Tento manuál obsahuje informace ohledně struktury projektového adresáře.

V hlavním adresáři projektu se nachází následující podadresáře: `config`, `fonts`, `lib`, `noCompileLibs`, `sounds`, `src`, `symbols`. Dále se v tomto adresáři také nachází soubor `Makefile`, `.gitignore`, DLL knihovny a vývojářská příručka pro operační systém Windows v souboru `README.md`.

Adresář `config` obsahuje konfigurační soubory pro všechny komponenty typu `Model`, dále pak simulační konfigurační soubory, které slouží pro ladění aplikace a napodobení komunikace s modulem `EVC`, konfigurační soubor s definicí barev a konfigurační soubor pro objekty typu klávesnice.

Adresář `fonts` obsahuje soubory s popisy počítačových písem.

Adresář `lib` obsahuje zdrojové soubory s funkcemi z knihovny od Petra Stříteského a také soubory s definicemi určitých aplikačních proměnných, výčtových typů a datových typů, které používá jak Petrova knihovna, tak i aplikace. Dále se zde nachází hlavičkový soubor knihovny `JSON for Modern C++`.

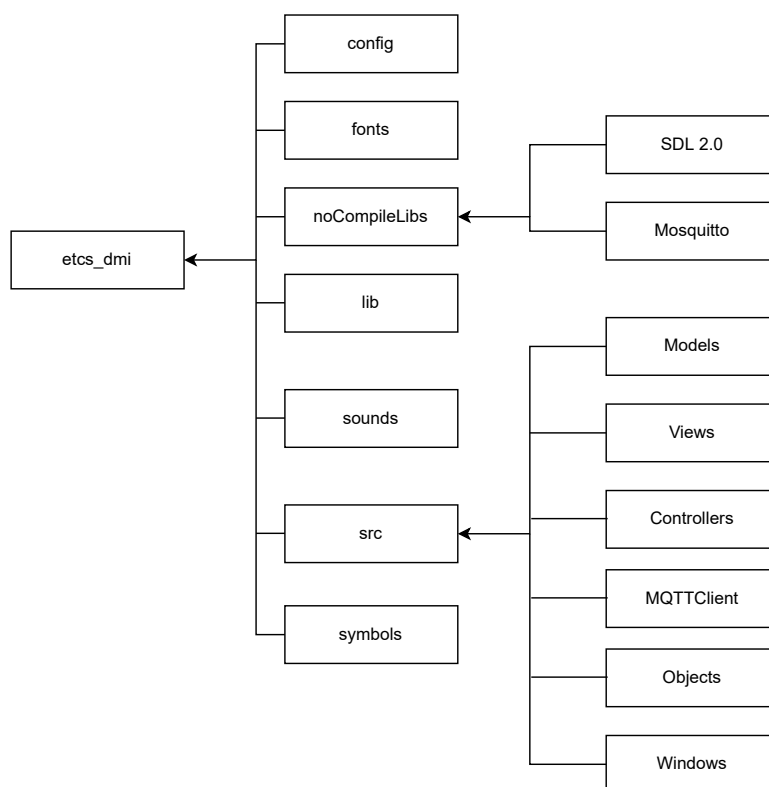
V adresáři `noCompileLibs` se nachází soubory knihoven `SDL 2.0` a `Mosquito`, které nejsou určeny ke kompilaci, ale slouží pouze pro importování do projektu ve `Visual Studiu`.

V adresáři `sounds` nalezneme soubory se zvuky, které aplikace využívá.

Adresář `symbols` zahrnuje bitmapové soubory s obrázky a symboly, které aplikace zobrazuje.

V adresáři `src` poté nalezneme zdrojové a hlavičkové aplikační soubory. Adresář zahrnuje komunikační třídy, třídu pro koordinační komponentu a také třídy využití při návrhovém vzoru `Observer`. Dále se zde vyskytují podadresáře pro obrazovky, objekty typu tlačítko a klávesnice, klientskou stranu `Mosquitta` a jednotlivé komponenty typu `Model`, `View`, `Controller`.

Pro přehlednější orientace mezi jednotlivými adresáři je zde umístěn adresářový diagram.

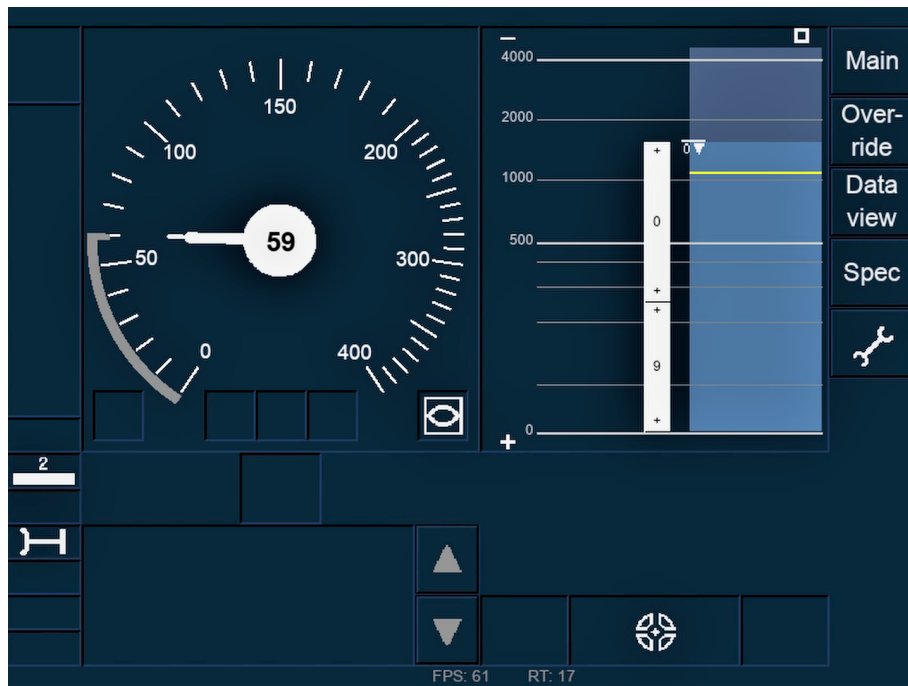


Obrázek 6.6: Adresářový diagram

## 6.4 Ukázka uživatelského rozhraní

Vzhled a funkcionality uživatelského rozhraní jsou dány dokumentací, nicméně zde přiblížím podobu výchozí obrazovky během dvou odlišných situací.

V následujícím obrázku je znázorněn průběh jízdy za normálních podmínek a povolené rychlosti 60 km/h, přičemž aktuální rychlost vlaku je 59 km/h. V plánovací oblasti můžeme vidět vzdálenost do konce oprávnění k jízdě, indikační značku a také dva gradienty znázorňující stoupání na trati.



Obrázek 6.7: Jízda za normálních podmínek

Druhý obrázek zobrazuje průběh jízdy v momentě, kdy strojvedoucí překročí nejen povolenou rychlost, ale i rychlost, při které dochází k zásahu nouzové/provozní brzdy. Z tohoto důvodu je ukazatel rychlosti zbarvený do červené barvy, přičemž kruhový ukazatel rychlosti je v rozmezí cílové rychlosti (zde 0 km/h, protože se jede na konec trati) až po povolenou rychlost ve žluté barvě a v rozmezí mezi povolenou rychlostí a rychlostí zásahu nouzové/servisní brzdy v červené barvě. Dále vidíme v levé části obrazovky vedle rychloměru vykreslenou vzdálenost k cíli, která je rovna 280 metrům.



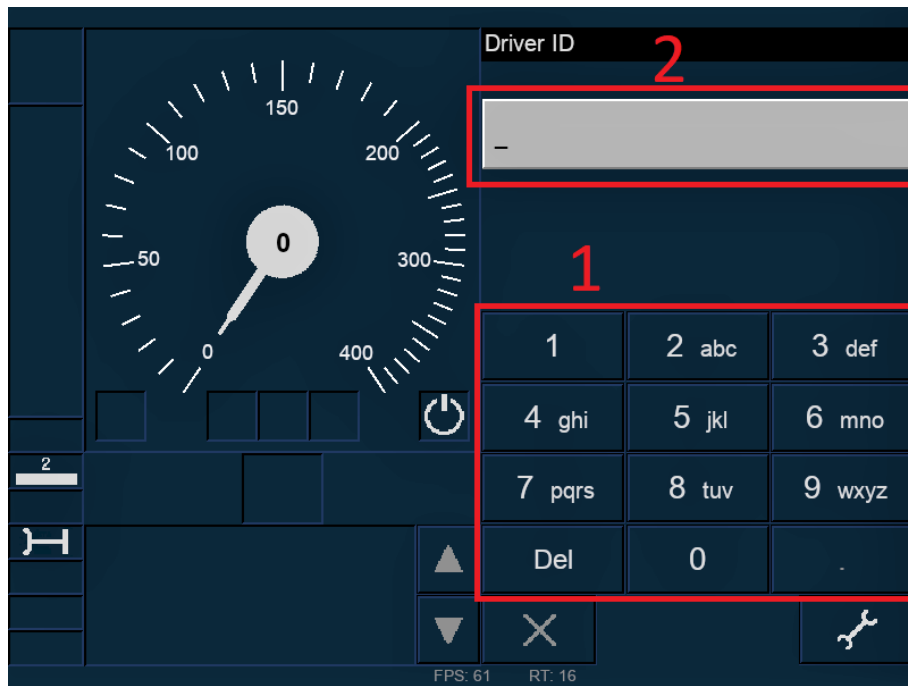
Obrázek 6.8: Jízda v kritické situaci

## 6.5 Uživatelská příručka

Uživatelská příručka poskytuje základní informace o ovládní aplikace pro běžného uživatele. Z důvodu zjednodušení je v této části popsáno ovládní a instrukce během procedury zahájení mise. Pro správný běh aplikace je nutné mít také spuštěný modul EVC, RBC a ovládací panel strojvedoucího.

### 6.5.1 Zahájení mise

Po spuštění aplikace DMI je zobrazena černá obrazovka. Prvním krokem je postupně zapnutí kabiny a řízení pomocí dvou příslušných spínačů na ovládacím panelu strojvedoucího. Tímto se spustí procedura zahájení mise a zobrazí se obrazovka čísla strojvedoucího. Zde pomocí alfanumerické klávesnice na obrazovce je nutné zadat číslo strojvedoucího (1). Po dokončení zadávání stačí pro potvrzení stisknout šedé pole pro vstup, ve kterém je zobrazeno zadané číslo (2).

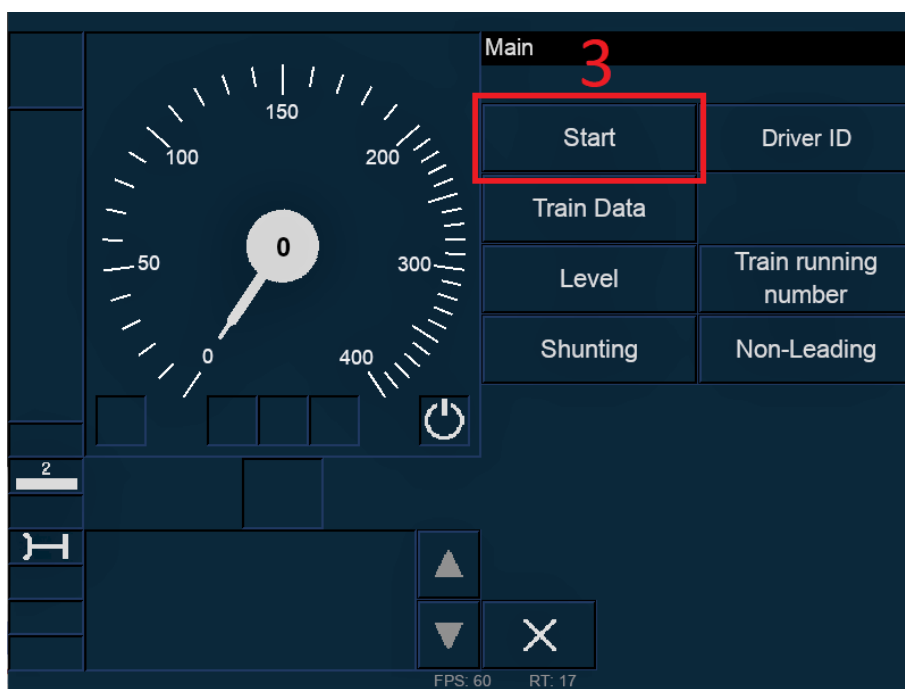


Obrázek 6.9: Obrazovka čísla strojvedoucího

Následně se zobrazí hlavní obrazovka s vypnutými tlačítky a pohyblivým symbolem přesýpacích hodin, značící čekání na spojení s modulem RBC. Po obdržení spojení s modulem RBC se zobrazí hlavní obrazovka se zapnutými tlačítky. Zde pro zahájení jízdy vlakem je nutné stisknout tlačítko Start (3) a vyčkat na obdržení oprávnění k jízdě. Ihned poté je možné začít pohyb vlaku pomocí ovládacího panelu strojvedoucího.

## 6. IMPLEMENTACE

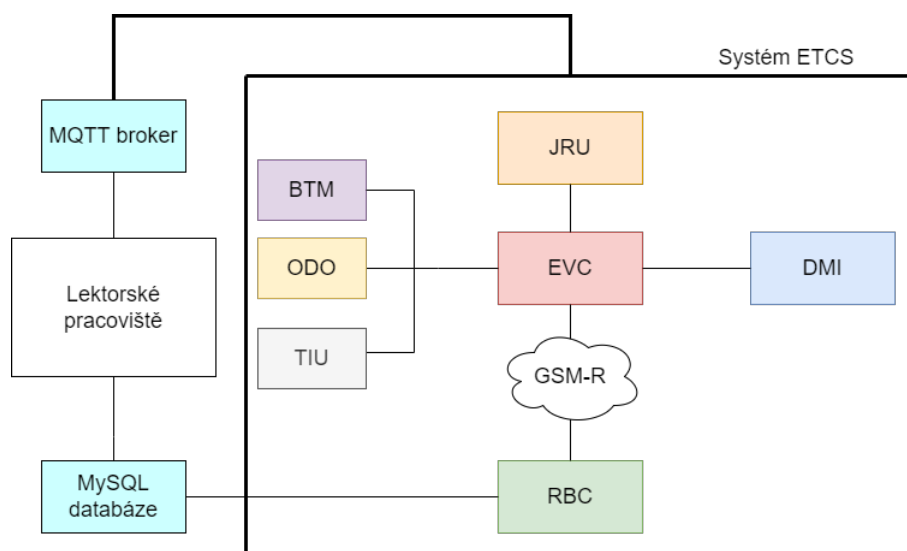
---



Obrázek 6.10: Hlavní obrazovka

## 6.6 Schéma vlakového simulátoru

V této části se nachází nadhled na vlakový simulátor s komponentou ETCS tvořený ve spolupráci s Fakultou dopravní ČVUT. Následující schéma znázorňuje vztah mezi jednotlivými komponentami tohoto simulátoru.



Obrázek 6.11: Schéma ETCS

### Lektorské pracoviště

Lektorské pracoviště „realizuje obsluhu průběhu simulace, nastavuje parametry simulace a provádí vyhodnocení průběhu simulace“. Viz [9].

### MQTT broker

Jedná se o MQTT server, obsahující data, která se vyměňují mezi jednotlivými moduly. Lektorské pracoviště a moduly systému ETCS se k němu připojují a potřebná data z něj odebírají nebo upravují jejich hodnoty. Funguje tedy jako datový sklad. Viz [9].

### MySQL databáze

V databázi MySQL je uložen popis traťové infrastruktury, jejíž data pak získává modul RBC právě z této databáze. Viz [9].

### GSM-R

Jedná se o „globální systém pro mobilní komunikace (GSM) pro železniční aplikace. Při aplikaci systému ETCS se uplatňuje zejména pro komunikaci

*mezi vlakem a infrastrukturou (RBC) v rámci ETCS aplikační úrovně 2 a 3<sup>a</sup>. Viz [9].*

### **Modul JRU**

*Modul JRU „zahrnuje datový záznamník na úrovni tzv. černé skříňky, který slouží v případě nehody nebo problematického chování mobilní části ETCS k analýze nastalého děje“. Viz [9].*

### **Modul BTM**

*Tento modul „zprostředkovává informaci o datech přečtených jednotkou BTM (Balise transmission module) ze systému EUROBalise (balíza umístěná v kolejišti)“. Viz [9].*

### **Modul TIU**

*„Jedná se modul (spíše chápáno jako rozhraní) realizující rozhraní k vlaku (Train interface unit)“. Viz [9].*

### **Modul ODO**

*„Tento modul (spíše chápáno jako rozhraní) zprostředkovává informace o poloze vlaku na základě informace z fyzikálního modelu vlaku“. Viz [9].*

### **Modul EVC**

*Modul EVC, propojený s ostatními komponentami systému ETCS, „realizuje hlavní výpočetní funkce mobilní části systému ETCS“. Viz [9].*

### **Modul RBC**

*Modul RBC představuje „infrastrukturní část, která musí zajišťovat vypracování komunikačních zpráv pro mobilní část EVC“, viz [9], zahrnující především informace o:*

- Statickém profilu tratě
- Omezujících podmínkách infrastruktury
- Povolení k jízdě (MA – oprávnění k jízdě)



## Modul DMI

Tento modul zobrazuje strojvedoucímu informace o jízdě a vlaku, přičemž zároveň funguje jako ovládací displej pro strojvedoucího. Komunikuje s komponentou EVC, se kterou si vzájemně vyměňují potřebná data.

V rámci přehledu je zde konkrétněji popsána práce mě a mých kolegů na aplikaci DMI. Kolegové implementovali následující funkční požadavky, které byly buď implementovány celé, nebo pouze z části (přehled funkcionalit a jejich aktuální kompletnosti se nachází v kapitole 4.1.1 v této práci): Komunikace s modulem EVC, Rychloměr, Kruhový ukazatel rychlosti, Brzdné informace, Doplnkové informace a Plánovací oblast.

Mnou implementované funkční požadavky v rámci rozsahu bakalářské práce jsou tyto: Procedura zahájení mise, Hlavní obrazovka, Obrazovka čísla strojvedoucího. Dále jsem nad rámec rozsahu pracoval na plánovací oblasti, rychloměru, kruhovém ukazateli rychlosti, monitorovacích informacích a funkci vypnutí/zapnutí zobrazení informací o rychlosti.



---

# Testování

Tato kapitola se věnuje testování prototypu aplikace, který je podroben uživatelským, akceptačním a integračním testům.

## 7.1 Uživatelské testování

Pro účely uživatelského testování jsou zde sepsány tři testovací scénáře, testující průchody prototypem aplikace a jeho interaktivní funkcionality. Tyto scénáře jsou určeny pro testování na hotovém simulátoru na Fakultě dopravní, který ale aktuálně bohužel není ještě dokončen. Z tohoto důvodu, po domluvě s vedoucím práce, k uživatelskému testování prozatím nedošlo.

### 7.1.1 Zadání čísla strojvedoucího a zahájení jízdy

#### Odhadovaný čas

Do 1 minuty.

#### Účel scénáře

Cílem je otestovat zadání čísla strojvedoucího a obdržení povolení oprávnění k jízdě v rámci procedury zahájení mise.

#### Počáteční bod

Prototyp zobrazuje obrazovku čísla strojvedoucího.

#### Koncový bod

Prototyp zobrazuje výchozí obrazovku.

## 7. TESTOVÁNÍ

---

### Instrukce

- Zadejte číslo strojvedoucího „admin123“.
- Zažádejte o povolení oprávnění k jízdě.

### Očekávané kroky

1. Zadání čísla strojvedoucího.
2. Potvrzení zadaného čísla stisknutím pole pro vstup v obrazovce čísla strojvedoucího.
3. Vyčkání na obdržení spojení s modulem RBC.
4. Stisknutí tlačítka Start.

### Kritické body

- Uživatel možná netuší, jak potvrdit zadané číslo strojvedoucího.
- Uživatel si nemusí být jistý, které tlačítko umožňuje zažádání o povolení oprávnění k jízdě.

### 7.1.2 Zobrazení obrazovky prohlížení dat a změna čísla strojvedoucího

#### Odhadovaný čas

2-3 minuty.

#### Účel scénáře

Cílem je otestovat zobrazení aktuálních vlakových dat v obrazovce prohlížení dat a následně změnit zadané číslo strojvedoucího. Poté je důležité zkontrolovat změnu při opětovném zobrazení vlakových dat.

#### Počáteční bod

Prototyp zobrazuje výchozí obrazovku po dokončené proceduře zahájení mise.

#### Koncový bod

Prototyp zobrazuje obrazovku prohlížení dat.

### Instrukce

- Změňte číslo strojvedoucího na „321admin“.
- Zkontrolujte, že se číslo strojvedoucího změnilo na novou hodnotu.

### **Očekávané kroky**

1. Zobrazení obrazovky prohlížení dat.
2. Zapamatování aktuální hodnoty čísla strojvedoucího.
3. Zobrazení obrazovky čísla strojvedoucího.
4. Zadání nové hodnoty čísla strojvedoucího.
5. Zobrazení obrazovky prohlížení dat.
6. Kontrola korektnosti zobrazené nové hodnoty čísla strojvedoucího.

### **Kritické body**

- Uživateli může trvat déle najít cestu z výchozí obrazovky do obrazovky čísla strojvedoucího.

### **7.1.3 Prohlížení plánovací oblasti**

#### **Odhadovaný čas**

1-2 minuty.

#### **Účel scénáře**

Cílem je otestovat korektnost funkcionality přiblížení/oddálení a schování/zobrazení logaritmicky škálované plánovací oblasti ve výchozí obrazovce.

#### **Počáteční bod**

Prototyp zobrazuje výchozí obrazovku se znázorněnými gradienty v plánovací oblasti.

#### **Koncový bod**

Prototyp zobrazuje výchozí obrazovku.

#### **Instrukce**

- Zobrazte celý profil gradientů i s viditelným koncem oprávnění k jízdě.
- Zobrazte pouze první gradient.
- Schovejte zobrazování plánovací oblasti.

### Očekávané kroky

1. Stisknutí tlačítka „+“ v plánovací oblasti, dokud se nebude zobrazovat celý profil gradientů i s koncem oprávnění k jízdě.
2. Stisknutí tlačítka „-“ v plánovací oblasti, dokud se nebude zobrazovat pouze první gradient.
3. Stisknutí tlačítka pro schování/zobrazení plánovací oblasti.

### Kritické body

- Uživatel nemusí hned objevit tlačítko pro schování/zobrazení plánovací oblasti.

## 7.2 Akceptační testy

Účelem akceptačních testů je ověření funkcionalit a kvality dodaného produktu, viz [10]. Tento druh testování provádí zadavatel, kterým je v případě této práce pan doc. Ing. Martin Leso, Ph.D. z Fakulty dopravní ČVUT. Dodaný prototyp byl zaměřen pouze na jízdu v módu FS, aby bylo možné co nejrychleji otestovat funkčnost jednotlivých částí.

Pan Leso dohlížel na zhotovené funkcionality a odsouhlasil implementovaný prototyp, který byl také porovnán, z hlediska korektnosti, s ERTMS Operational simulátorem od firmy Clearsky na Fakultě dopravní. Při totožných situacích se pak prototyp choval stejně. Následující tabulka obsahuje funkční požadavky, které byly schváleny v daném stavu.

Tabulka 7.1: Tabulka schválených funkčních požadavků

<b>Funkční požadavek</b>	<b>Stav</b>
F1 – Komunikace se strojvedoucím	Kompletní
F2 – Komunikace s modulem EVC	Kompletní
F26 – Procedura zahájení mise	Nekompletní, nejsou zhotoveny všechny potřebné obrazovky
F3 – Rychloměr	Změna barvy ručičky pouze v určitých stavech dohledu
F4 – Kruhový ukazatel rychlosti	Zobrazování rychlostí pouze v určitých stavech dohledu
F5 – Brzdné informace	Chybí indikátor zásahu nouzové/provozní brzdy a vypnutí/zapnutí zobrazení informací o rychlosti
F6 – Doplnňkové informace	Chybí informace o volné koleji před vlakem, textové zprávy, příkazy a oznámení o stavu trati, informace o úrovni jsou nekompletní
F7 – Plánovací oblast	Chybí příkazy a oznámení
F8 – Monitorovací informace	Chybí indikace povolení couvání, lokální čas a geografická pozice vlaku
F9 – Výběr tlačítek pro přepínání podúrovňových obrazovek	Kompletní
F10 – Hlavní obrazovka	Kompletní
F16 – Obrazovka čísla strojvedoucího	Kompletní

Po dodání prototypu jsem pokračoval v implementaci dalších funkcionalit, které prozatím nebyly tímto způsobem otestovány. Přehled aktuální rozpracovanosti se nachází v kapitole 4.1.1.

### 7.3 Integrační testy

Integrační testy slouží pro ověření celkové funkčnosti všech modulů a komponent dohromady, viz [10]. Zde jsou zaměřené na testování korektního zpracování a zobrazení dat od modulu EVC.

V DMI jsou tyto testy realizované pomocí odebírání dat ze souboru ve formátu JSON, který obsahuje několik po sobě jdoucích struktur, jejichž podoba je stejná jako struktury dat přicházející od skutečného modulu EVC. Tímto způsobem je v aplikaci namockované EVC a nasimulována komunikace.

Struktury jsou načítány postupně s volitelným časovým odstupem, přičemž jakékoliv chyby ve zpracování jsou odchyťovány do logovacího souboru. Ihned po načtení se tyto hodnoty dat zobrazí, čímž je umožněné otestování správného fungování zobrazování příslušných dat a tedy i fungování jednotlivých komponent v aplikaci.



---

## Závěr

Cílem mé práce bylo navrhnout, implementovat a otestovat prototyp aplikace DMI displeje, který je součástí dopravního simulátoru s komponentou ETCS. Podkladem pro dosažení cíle byla dokumentace k systému ETCS se zaměřením na modul DMI ve verzi 2.3.0, která byla v první kapitole analyzována. Prototyp byl implementován s použitím grafické knihovny od studenta Fakulty dopravní ČVUT, Petra Stříteského, jehož diplomová práce zde byla s ohledem na knihovnu také analyzována.

Na základě nabytých poznatků z analýzy dokumentace a konzultace se zadavateli z Fakulty dopravní ČVUT vzešly funkční a nefunkční požadavky pro kompletní aplikaci. V rámci návrhové kapitoly byla představena aktuální architektura prototypu aplikace včetně komunikace s modulem EVC.

Dle návrhu byl prototyp následně implementován v jazyce C++, přičemž po konzultaci s vedoucím práce, z důvodu rozsahu a komplexnosti celé aplikace, jeho výsledná verze neobsahuje všechny funkcionality definované ve funkčních požadavcích. Prototyp je funkční na operačních systémech Windows a Linux Debian 11.3.0. Na průmyslovém displeji funkčnost prototypu ověřena nebyla, protože bohužel ještě nebyl připravený pro zprovoznění aplikace.

Nakonec byl prototyp podroben akceptačním a integračním testům. Uživatelské testování po konzultaci s vedoucím práce, z důvodu nedokončeného vlakového simulátoru, nebylo uskutečněno, nicméně jako podklad pro jeho budoucí provedení slouží připravené testovací scénáře.

Ze zadání práce tedy byly kromě uživatelského testování a funkčnosti na průmyslovém displeji splněny všechny body. Ke zprovoznění aplikace a jejímu dalšímu vývoji slouží vytvořené příručky.

Pro budoucí rozšíření aplikace je v první řadě důležité doimplementovat všechny chybějící funkcionality, které nebyly zhotoveny. Dále je poté možnost aplikaci rozšiřovat a upravovat v rámci požadavků dokumentací k DMI displeji ve verzích 3.4.0 a 3.6.0, které jsou aktuálnější a obsahují určité změny i nové funkcionality.



---

## Literatura

1. EUROPEAN UNION AGENCY FOR RAILWAYS. *ETCS Driver Machine Interface* [online]. 2009. Ver. 2.3 [cit. 2021-12-20]. Dostupné z: [https://www.era.europa.eu/content/informative-set-specifications-1-etcs-b2-gsm-r-b1\\_en](https://www.era.europa.eu/content/informative-set-specifications-1-etcs-b2-gsm-r-b1_en).
2. STRÍTESKÝ, Petr. *Generický návrh provozních displejů drážních vozidel*. 2021. Diplomová práce. České vysoké učení technické v Praze, Fakulta dopravní.
3. KADLČEK, David; STEJSKAL, Jan; JAHODA, Petr; UDAVICHENKA, Yury; MACHÁČEK, Jiří; VEJVODA, Štěpán. *Návrh projektu DMI displej pro simulátor ETCS* [online]. 2021. Ver. 0.1.1 [cit. 2022-02-07]. Dostupné z: <https://drive.google.com/file/d/1z36wVkJQNMtTMDwFy7yTjKPnNGgGuXCMf/view?usp=sharing>.
4. LIGHT, Roger. *MQTT man page* [online]. Eclipse Foundation, Inc. [cit. 2022-02-07]. Dostupné z: <https://mosquitto.org/man/mqtt-7.html>.
5. LOHMANN, Niels. *JSON for Modern C++* [online] [cit. 2022-02-08]. Dostupné z: <https://json.nlohmann.me/>.
6. STRÍTESKÝ, Petr. *Generické knihovny pro návrh a vývoj provozních obrazovek kolejových vozidel v jazyce C++ s využitím knihoven SDL2.0*. 2021.
7. LAZY FOO' PRODUCTIONS. *Setting up SDL 2 on Linux* [online]. 2015 [cit. 2022-02-20]. Dostupné z: [https://lazyfoo.net/tutorials/SDL/01\\_hello\\_SDL/linux/index.php](https://lazyfoo.net/tutorials/SDL/01_hello_SDL/linux/index.php).
8. MACHÁČEK, Jiří; STEJSKAL, Jan. *Návod pro zprovoznění projektu* [online]. 2022 [cit. 2022-02-25]. Dostupné z: [https://drive.google.com/file/d/1o95BNVWDJJ6HwVlfjtiz8PCs2nXj0ok\\_/view?usp=sharing](https://drive.google.com/file/d/1o95BNVWDJJ6HwVlfjtiz8PCs2nXj0ok_/view?usp=sharing).
9. LESO, Martin. *Koncepce vlakového simulátoru ČVUT – komponenta ETCS*. 2021. Ver. 2.

## LITERATURA

---

10. KITNER, Radek. *Typy testování software (třídění testů)* [online] [cit. 2022-04-16]. Dostupné z: [https://kitner.cz/testovani\\_softwaru/typy-testovani-software-trideni-testu/](https://kitner.cz/testovani_softwaru/typy-testovani-software-trideni-testu/).

## Seznam použitých zkratk

### A.1 Zkratky k ETCS

**BTM** Balise transmission module

**CSM** Ceiling speed monitoring

**DMI** Driver machine interface

**EOA** End of authority

**ERTMS** European Rail Traffic Management System

**ETCS** European Train Control System

**EVC** European vital computer

**FS** Full supervision

**GSM-R** Global System for Mobile Communication - Railway

**JRU** Juridical recording unit

**ODO** Odometry

**PIM** Pre-indication speed monitoring

**RBC** Radio block centre

**RSM** Release speed monitoring

**SR** Staff responsible

**TIU** Train interface unit

**TSM** Target speed monitoring

## A.2 Ostatní

**BI-SP1** Softwarový týmový projekt 1

**BI-SP2** Softwarový týmový projekt 2

**JSON** JavaScript Object Notation

**MQTT** Message Queuing Telemetry Transport

**MVC** Model-View-Controller

**MySQL** My Structured Query Language

**RGB** Red green blue

**SDL** Simple DirectMedia Layer

**TCP/IP** Transmission Control Protocol/Internet Protocol

## Obsah přiloženého CD

	readme.txt .....	stručný popis obsahu CD
	exe .....	adresář se spustitelnou formou implementace na OS Windows
	src	
	impl .....	zdrojové kódy implementace
	thesis .....	zdrojová forma práce ve formátu $\text{\LaTeX}$
	text .....	text práce
	text.pdf .....	text práce ve formátu PDF