



Fyzicky realistický simulátor pražských tramvají

Ročníková práce

Ročníková práce

Fyzicky realistický simulátor pražských tramvají včetně okolí bodovacího systému

Gymnázium, Praha 6, Arabská 14

Arabská 14, Praha 6, 160 00

ROČNÍKOVÁ PRÁCE

Předmět: Fyzika

Téma: Fyzicky realistický simulátor pražských tramvají včetně okolí bodovacího systému

Autor: Martin Bykov

Třída: 2.E

Školní rok: 2024/2025

Vedoucí práce: Mgr. Jan Tuzar

Třídní učitel: Mgr. Jan Tuzar

Prohlášení

Prohlašuji, že jsem jediným autorem tohoto projektu, všechny citace jsou řádně označené všechna použítá literatura další zdroje jsou práci uvedené.

Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů uděluji bezúplatně škole Gymnázium, Praha 6, Arabská 14 oprávnění výkonu práva na rozmnožování díla (§ 13) práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou bez omezení územního rozsahu.

V Praze dne __. března 2025

X

Martin Bykov, 2.E

Poděkování

Zde bych rád poděkoval panu Eduardovi Kastnerovi, za poskytnutí informací o vysvětlení specifik řízení ovládání tramvaje a panu Martinovi Leubnerovi, za pomoc při nahrání zvuků během jízdy noční tramvaje. Oběma předem zmíněným lidem bych také rád poděkoval za pomoc při testování simulátoru. Dále bych rád poděkoval panu Martinovi Zahrádkovi za pomoc při hledání referenčního materiálu pro modelování tramvají.

Obsah

Seznam použitých zkratk odborných termínů.....	7
Úvod.....	11
1. Vozidla.....	12
1.1. Modelování vozidel.....	12
1.1.1. T3R.P.....	12
1.2. Texturování nátěry vozidel.....	12
1.2.1. T3R.P.....	12
1.3. Konfigurace vozidel.....	16
2. Implementace vlastní grafické infrastruktury.....	20
2.1. Okno grafické rozhraní.....	20
2.1.1. Postup implementace.....	20
2.1.2. Kompatibilita mezi operačními systémy.....	20
2.1.3. Infrastruktura vykreslování objektů na obrazovku.....	21
2.1.4. Grafické rozhraní.....	22
2.2. Vykreslování modelů.....	23
2.2.1. Načítání vykreslování modelů, systém GMS.....	23
2.2.2. Načítání vykreslování textur modelů.....	24
2.2.3. Vykreslování průhledných částí modelů.....	24
2.3. Vykreslování okolí.....	25
2.3.1. Vykreslování kolejí.....	25
2.3.2. Vykreslování silnic, země, ulic, kolejových svršků stěn.....	26
2.3.3. Vykreslování budov památek.....	26
3. Prvky předpisů ovládání simulátoru.....	27
3.1. Pohyb po trati návěsti.....	27
3.2. Výhybky.....	28
3.2.1. Elektrické.....	28
3.2.2. Rádiové.....	28
3.2.3. Zajišťovací výhybky.....	28
3.2.4. Přehazování výhybek.....	28
3.3. Návěstidla.....	29
3.3.1. Výhybková návěstidla.....	29
3.3.2. Křižovatková návěstidla.....	30
3.3.3. Předvěsti křižovatek křižovatek návěstidel.....	30
3.4. Jízdní řád.....	31
3.4.1. Normální zastávky.....	31
3.4.2. Zastávky na znamení.....	31
3.4.3. Kontrolní body.....	31
3.4.4. Konečné zastávky.....	31
3.5. Identifikace tramvaje.....	32
4. Dodatečné nástroje formáty jejich souborů.....	33
4.1. Nástroj na úpravu mapy a scénářů.....	33
4.1.1. Základní informace.....	33
4.1.2. Horní lišta.....	34

4.1.3. Menu úprav.....	35
4.1.4. Plátno úprav.....	35
4.1.5. Tvorba linky.....	36
4.2. Formát souboru okolí.....	38
4.3. Formát souboru scénáře.....	42
4.4. Nástroj na vytváření hlásiče.....	44
4.5. Formát souboru hlásiče.....	45
5. Implementace simulací.....	47
5.1. Infrastruktura působení sil na vozidla.....	47
5.2. Tíha.....	47
5.3. Tření, valivý odpor a prokluz kol.....	48
5.4. Síla působení motorů a rychlost vozidla.....	49
5.5. Síly v oblouku a vykolejení.....	50
5.6. Aerodynamická odporová síla.....	51
5.7. Elektrické systémy vozidel celková práce vozidla.....	51
5.8. Osvětlení.....	52
5.9. Stíny.....	55
5.10. Simulace zvuků.....	57
5.10.1. Simulace zvuku motoru a brzd.....	57
5.10.2. Simulace zvuku jízdy.....	57
5.10.3. Simulace jiných zvuků.....	57
5.10.4 Simulace hlášení.....	57
5.11. Výpočet srovnávání emisí.....	58
5.11.1. Výpočet emisí automobilové dopravy.....	58
5.11.2. Výpočet emisí tramvaje.....	58
5.11.3. Výpočet ušetřených emisí.....	58
6. Simulace počasí a jeho účinků.....	59
6.1. Infrastruktura systému počasí.....	59
6.2. Denní noční cyklus.....	59
6.3. Počasí.....	59
6.3.1. Vítr.....	59
6.3.2. Déšť.....	59
6.3.3. Bouřky.....	59
6.3.4. Mlha.....	59
6.4. Počasí závisující na ročním období.....	60
6.4.1. Sníh.....	60
6.5. Roční období.....	60
6.5.1. Jaro a léto.....	60
6.5.2. Podzim.....	60
6.5.3. Zima.....	60
7. Modelování okolí.....	61
7.1. Zastávky.....	61
7.2. Standardní budovy.....	61
7.2.1. Typy budov.....	61
7.2.2. Systém obarvení budov.....	64

8. Systém bodování.....	65
8.1. Seznam akcí za které se body odečítají.....	65
8.2. Seznam akcí za které se body přičítají.....	65
8.3. Výpočet výsledné známky.....	65
9. Další rozvoj projektu.....	67
9.1. Modelování dalších vozidel.....	67
9.2. Modelování dalších tramvajových tratí.....	67
9.3. Vylepšení modelu osvětlení.....	68
9.4. Přidání řek a mostů.....	68
9.5. Přidání opravdového terénu.....	68
9.6. Přidání nastavení trakčního vedení a stahování pantografu.....	68
Závěr.....	69
Použité technologie.....	70
Seznam obrázků.....	71
Seznam tabulek.....	71
Citace.....	72

Seznam použitých zkratk odborných termínů

ASCII Kódování znaků zaujímající vždy právě 1 byte. Podporuje pouze anglickou abecedu, číslice, pomocné znaky typu mezer závorek či kontrolní znaky.

Abstrakce Zabalení obvykle několika systémových („nízko-úrovňových“) funkcí do jiné funkce či objektu za účelem lehčího vývoje použití daných funkcí.

Anti-aliasing Technologie zahlazování pixelu pro zahlázení viditelného přechodu mezi pixely

APEX Společnost vyvíjející palubní počítače, hlásiče a rádiové stanice. Jejich název se používá jako přezdívka palubního počítače tramvaje řad T3R.P, T3R.PV, T3R.PLF, T6A5 KT8D5.RN2P.

API Application programming interface – soubor funkcí a tříd pro přístup programu nějaké infrastruktuře a funkcím.

Atenuace Zeslabení vlnění při vzdálenosti (při průchodu jakéhokoliv materiálu). grafickém programování znamená hodnotu zeslabení světla určitou vzdáleností.

Bitové pole Místo uložení, zpravidla několika, Booleanových hodnot. Bitové pole ukládá hodnoty které mají vliv na pracování programu, jinými slovy flagy.

Boolean Hodnota která může nabývat pouze hodnoty 0 či 1 a je jí možné uložit do 1 bitu paměti.

Buffer Název sloví OpenGL pro úložný prostor aneb vyhrazená paměť pro ukládání dat na grafické kartě. Existuje vícero typů bufferů (vertexový, indexový, shaderový, framebufferový apod.).

DPP Dopravní podnik hlavního města Prahy (také známý jako Dopravní podnik hl. m. Prahy, a. s.), jediný provozovatel tramvají typu T3R.P v Praze.

Dvojkový doplněk Standardní způsob ukládání záporných hodnot binární reprezentací celých čísel. První bit určuje hodnotu jako -2^{bits} kde je *bits* počet bitů čísla.

Endianita, little-endian, big-endian Pořadí bytů při ukládání čísla. Ve formátu little-endian je na konci byte nejmenší hodnota. Ve formátu big-endian je na konci byte největší hodnotou. Endianita (*angl. Endianness*) určuje jaký z těchto dvou formátů je použit.

ev. č. Evidenční číslo vozidla, užívané pro identifikaci vozidla dopravcem.

Flag Booleanova hodnota mějící vliv na průběh programu.

Fragment Pixel, bod ve světě – pro každý takový bod je spuštěn tzv. fragmentový shader, který vypočítává jeho barvu.

Framebuffer Speciální buffer s napojenou texturou použitý pro vykreslování mimo obrazovku.

GLAD Knihovna pro načítání funkcí všech variant API OpenGL.

GLFW Knihovna pro vytváření okna a detekce vstupu z klávesnice či myši. Plným názvem OpenGL Framework.

GLM Knihovna pro práci se strukturami, jako například vektory či matice, často v OpenGL užíváné.

GLSL Jazyk, podobný jazyku C avšak s zmenšeným počtem funkcí, použitý v shaderech OpenGL.

Hexadecimální kód Způsob kódování RGB barev ve rozsahu od 0 až 255 (tzv. 8-bitové kombinování, $2^8 - 1 = 255$). Hodnoty jsou vyjádřené šestnáctkové soustavě (od 00 do ff).

Index Identifikační číslo vertexu. Jsou používány pro eliminaci duplicitních vertexů. Místo duplikování bude tak použit jeden vertex, na který se bude prostřednictvím indexu ukazovat.

Instancování Anglicky „instanced rendering“, též instancing neboli „instancování“. Jedná se o způsob vykreslování stejného meshe několikrát jediným příkazem nastavením potřebné proměnlivé informace (posunem, barvou apod.) za pomoci UBO nebo SSBO bufferů před vykreslováním. V shaderu je přístupná informace který objekt se jedná.

ISO Mezinárodní organizace pro normalizaci („International Organization for Standardization“) je organizací která stanovuje jasně daná pravidla pro, mimo jiné, barvy a formy varovných symbolů.

IZS Integrovaná záchranná služba. V tomto textu je zkratka použita jako označení pro jakékoliv pohotovostní vozidlo, a to jak policejní, hasičské či nemocniční.

Leaflet Knihovna vytvořená v jazyce JavaScript používaná pro jednoduché vykreslování mapových podkladů.

Materiál Soubor vlastností pro daný mesh. Mezi vlastnosti patří například textura, barva, lesk, odrazivost, průhlednost atd.

Mesh Odborný termín pro veškerá data (vertexy, obličeje) v modelu nebo jeho části. Nejčastěji souvislý.

MHD Zkratka pro městskou hromadnou dopravu.

MSAA „Multisample anti-aliasing“ – anti-aliasing technologie poskytnutá API OpenGL.

Normalizace Normalizace vektoru je převedení vektoru takovým způsobem, aby udával stejný uhel a jeho délka činila hodnotu 1.

Normál Vektor kolmý na daný obličej.

Nulový znak Jinak známý jako nulový terminátor, tento znak o hodnotě 0 (0 neoznačuje číslici, nýbrž hodnotu bytu) často slouží jako zakončení textových řetězců.

OpenGL Zkrácená forma Open Graphics Library, API použité pro infrastrukturu simulátoru zabývající se vykreslováním objektů na obrazovku.

OpenStreetMap Zdarma open-source mapa, použita v nástrojích simulátoru.

PID Pražská integrovaná doprava, systém vytvořený organizací ROPID (Regionální organizátor PID) který je zodpovědný, mimo jiné, za nový jednotný nátěr vozidel.

Průhlednost V tomto textu se průhlednost udává jako viditelnost procentech, kde 0 procent znamená, že je daný objekt neviditelný zatímco 100 procent znamená to, že je zcela neprůhledný.

Pixel Nejmenší jednotka v počítačové grafice – označuje jedno světlo na monitoru.

RGB Název formátu kódování barev – znamená „Red, Green, Blue“.

Shader Program napsaný v jazyce GLSL spuštěný na grafické kartě během procesu vykreslování. Je používán pro výpočet a určení barev pixelů, změn pozic bodů či jiných výpočtů které je potřeba udělat pro každý pixel zvlášť.

STBI Knihovna pro načítání obrázků v kolekci jednosouborových knihoven STB. Plným názvem „STB Image“.

SSBO, UBO, Uniformy Prostředky pro komunikaci shaderu a programu. Uniformy mají nejmenší kapacitu, uniformová úložiště (UBO, „Uniform buffer object“) větší a SSBO („Shader storage buffer object“) největší.

Teselace neboli „parketování“ – pokrytí povrchu různými geometrickými tvary.

Textura Obrázek či jeho část určující (nebo ovlivňující) výslednou barvu jakéhokoliv fragmentu.

Texturové parcely Název pro samostatný obdélník s daným materiálem pro který jsou v programu vygenerovány vertexové a indexové buffery. Texturové parcely nemusí povinně mít textury, stejně jako materiály mohou mít pouze barvu.

Unixový čas Počet sekund uplynulých od 1. ledna 1970, 00:00:00 v koordinovaném světovém čase UTC.

UTF-8 Písmenná sada obsahující celou latinskou abecedu, číslice a většinu znaků s diakritikou užívanou v českém jazyce. Každé písmeno je uloženo v 1 bytu neboli 8 bitech.

Vertex Bod jakéhokoliv modelu. Může mít víc než jen polohovou informaci – může také obsahovat barvu, číslo materiálu, normály obličeje kterého je součástí atd.

Vykreslovací příkaz Příkaz který program posílá grafické kartě. Hlavní cíl jakékoliv optimalizace je minimalizace jeho počtu.

WAV Zkratka pro Waveform Audio File Format. Jedná se nekomprimovaný formát zvukových souborů.

Z-Boj (angl. *Z-fighting*) Neurčitost mezi dvěma texturami které jsou ve stejném místě, výsledek tohoto je problikávání obou textur na zasaženém místě.

Anotace

Cílem tohoto projektu je implementace realistického simulátoru pražských tramvají za pomoci jazyka C++ a grafického rozhraní OpenGL. Jsou simulované interakce vozidla a kolejí, rychlostní limity, výhybky, osvětlení, elektrické systémy tramvaje, efekt počasí a ročních období na předem zmíněné prvky a i realistický dynamický zvukový systém při rozjezdu a zatáčení. Současně je simulován pouze typ T3R.P na lince 13 v úseku Olšanské hřbitovy – Museum – Čechovo náměstí a odbočce na obratiště Zvonářka.

Abstract

The goal of this project is the implementation of a physically realistic simulator of trams in Prague using the C++ programming language and the OpenGL graphics API. Among the simulated parts are the interactions of the vehicle and the track, the speed limits, switches, lighting, the electrical systems of the tram, the effect of the weather on the mentioned parts as well as realistic dynamic sound system during the acceleration and turning of the tram. Currently, only the T3R.P tram type is simulated and only on line 13, which goes from Olšanské hřbitovy to Čechovo náměstí via Museum and the branch line to the Zvonářka terminus.

Аннотация

Цель этого проекта это создание физически реалистического симулятора пражских трамваев с помощью языка C++ и графической «библиотеки» OpenGL. Симулированные будут влияния трамвая и рельс, освещение, электрические системы трамвая, влияние погоды на все симуляции и симуляция звуков трамвая. В проекте симулирован только 1 тип трамвая — ТЗР.П — на линии 13 между остановкой Olšanské Hřbitovy и остановкой Čechovo náměstí вместе в ответвлении к остановке Zvonářka.

Úvod

Cílem tohoto projektu je implementace fyzicky realistického simulátoru pražských tramvají (současně simulace obsahuje pouze tramvaj T3R.P) za pomoci vlastní grafické infrastruktury. Simulátor obsahuje simulaci vzájemného působení (tření, zvuk) mezi kolejemi a koly tramvaje, simulaci výhybek, simulaci deformace silničních vozidel při nárazu, simulaci elektrických systémů tramvaje i vliv počasí na předem zmíněné prvky simulace. Projekt bude možné použít nejen jako simulátor pro trénování dalších řidičů tramvaje, ale vizualizace simulátoru bude možné také použít jako fyzikální znázornění simulovaných prvků.

1. Vozidla

1.1. Modelování vozidel

1.1.1. T3R.P

Modelování vozidla T3R.P bylo poměrně nenáročné. Modelování karoserie a formy tramvaje bylo jednoduché z důvodů kvalitního referenčního materiálu, nalezeného na fotogalerii vozidel MHD^{[3][4][5]}. Nejtěžší částí pro modelování byly podvozky tramvaje a pantograf (sběrač elektrického proudu z trakčního vedení) z důvodů nedostatku kvalitního zdrojového materiálu pro právě tyto části vozu.^[6] Pro pantograf byl později nalezen technický výkres, bohužel avšak pro původní, nerekonstruovanou řadu T3^[7], a pro podvozek byl později nalezen 3D model podobného typu T3.SUCS.^[8]

Řada T3R.P byla vybrána z důvodu její rozšířenosti ve flotile DPP, kde je 337 vozů řady T3R.P z 782 tramvají celkem, což celkem tvoří 43% procent všech tramvají.^[9]

1.2. Texturování nátěry vozidel

Texturování vozidla probíhalo ve 2 fázích. Nejdříve byli vytvořeny jednotlivé materiály a textury pro které byly následně různé části modelu přiřazeny. Piktogramy a varovné značky byly vytvořeny a přidány zvlášť. Pro zachování realismu byli mezinárodně standardizované varovné znaky ISO importovány z externích zdrojů.^{[10][11]}

1.2.1. T3R.P

Tramvaj v simulátoru má několik různých nátěrů. Jsou přidány oba dva běžné nátěry vozů DPP, to červeno-krémový a vinovo-stříbrný (užívaný je sice primárně vozy typu T3R.PLF, je ale také použit vlečnými vozy typu T3R.P a je možné zahlédnout tyto vozy samostatně během testovacích či zvláštních jízd). Všechny nátěry sdílí společné zabarvení interieru.^[3]

Hodnot barev schémat byli převzaty z veřejně dostupných fotografií tramvají typu T3R.P.^[3] Barvy pro moderní nátěr PID byly uvedeny v grafickém manuálu jednotného vizuálního stylu PID z roku 2022.^[12] V následující tabulce (*Tabulka 1*) jsou uvedeny názvy hexadecimálních RGB kódů všech barev přítomných na modelu tramvaje.

Do simulátoru je možné přidat také vlastní nátěry. Toto se dělá přidáním řádku do konfiguračního souboru.

Červená (nátěr)	#e72e19
Krémová (nátěr)	#f2ebdf
Vínová (nátěr)	#701021
Stříbrná (nátěr)	#7f7d7e
Červená, jednotný PID (nátěr)	#bb1e10 / RAL 3020
Šedá, jednotný PID (nátěr)	#b0b0a9 / RAL 7038
Hnědá (nátěr)	#6b5646
Bílá (interiér, laminát – světlejší části)	#eeeeee
Šedá (interiér, laminát – tmavší části), Železná, Skleněná	#aaaaaa
Tmavě šedá (interiér – šedé železné části)	#303030
Tmavě železná	#454545
Červený plast na sedačkách	#ff0000
Šedý plast na sedačkách	#454545
ISO 3864 – Žlutá (varování)	#f9a900
ISO 3864 – Červená (zákaz, požární technika)	#9b2423
ISO 3864 – Bílá (podklad)	#ecece7
Žlutá (tlačítka, tyče pro držení)	#ffce00
Oranžová (stroje pro označení jízdních dokladů)	#ff9000
Zelená (dveřní tlačítka)	#00c16e
Tmavě červená (lampy, zadní reflektory)	#800000
Tmavě oranžová (světla pro odbočování)	#a15a00
Černá	#000000
Modrá	#000080

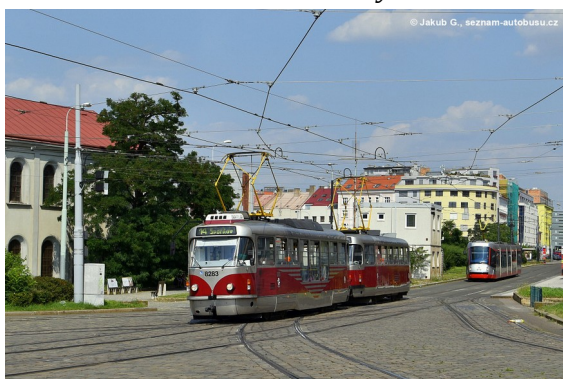
Tabulka 1: Seznam barev použitých v modelu.

Textury pro nerovnosti interiéru vozidla byly vytvořeny spojením šedého pozadí a vrstvy 90 procentní průhlednosti, na kterou byla následně nanесena kombinace šedé barvy a barevného elektronického šumu, který byl před aplikací převeden na černobílou škálu. Nerovnosti v textuře a interiéru byly nanесeny speciálním štětcem v editačním programu Krita pod názvem „Texture Crackles“. Textury pro světelná zařízení byly vytvořeny nastavením 70 procentní průhlednosti na požadovanou barvu. Pro čiré sklo byla použita železná barva, to s 90 procentní průhledností.



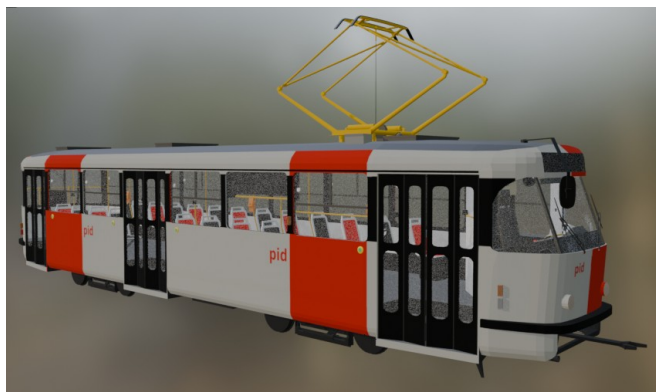
Obrázek 1: Fotografie vozů T3R.P (ev. č. 8234 8235) DPP nedaleko zast. Sídliště Řepy^[3]

Obrázek 2: Červeno-krémový nátěr na modelu tramvaje typu T3R.P



Obrázek 3: Vůz T3R.PLF (ev. č. 8283) a vlečný vůz T3R.P (ev. č. 8160) na Palmovce^[3]

Obrázek 4: Vínovo-stříbrný nátěr nanесený na modelu typu T3R.P



Obrázek 5: Umělecké zobrazení nátěru PID vytvořeného dle grafického manuálu PID

Ročníková práce

Fyzicky realistický simulátor pražských tramvají včetně okolí bodovacího systému

1.3. Konfigurace vozidel

Vozidla je možné přidávat a konfigurovat za pomoci konfiguračního souboru, umístěného ve složce programu. V konfiguračním souboru jsou uvedeny nejen fyzikální parametry vozidla – rozměry skříně vozidla ze kterých je pak vypočtena plocha stěn skříně vozidla, která je pak použita při výpočtech síly větru, odkaz na zvuky jízdy vozidla, hmotnost vozidla a kapacita vozu, ale také grafické parametry – barva panelů, seznam nátěrů, propojení meshů a modelů podvozků které je posouvají, posunu kamery řidiče či prvky ovládací – posun a rozměr tzv. „tlačítkových zón“, neboli zón při kliknutí na které se daná akce spustí. Každá hodnota je uložena následujícím formátu:

`NÁZEV_HODNOTY=HODNOTA`

NÁZEV_HODNOTY i *HODNOTA* jsou řetězce ve formátu UTF-8. Řetězce smí obsahovat všechny znaky UTF-8 kromě znaku „rovná se“, čárky středníku. Mezery jsou započítány jako součást hodnoty. Na každém řádku je přítomna právě jedna hodnota. Má-li konfigurační soubor pro nějaký parametr více hodnot, jsou jednotlivé hodnoty oddělené čárkou. Obsahuje-li parametr více vnitřních parametrů, jsou odděleny středníkem. Příklady formátování několikánásobných hodnot souboru jsou uvedeny zde:

`NÁZEV_HODNOTY=HODNOTA, HODNOTA`

`NÁZEV_HODNOTY=HODNOTA, HODNOTA; HODNOTA2, HODNOTA2`

V následujících tabulkách jsou uvedeny všechny povinné hodnoty, jak pro vůz T3R.P^[13], tak i obecné konfigurační hodnoty. V tabulkách nejsou uvedeny hodnoty souborových cest, jelikož jsou relativní a mohou se jednoduše měnit. Konfigurační soubor musí povinně všechny uvedené hodnoty obsahovat. Souborové cesty jsou uloženy k souborům ukládajícím zvuky rozjezdu a brzdění, jízdy, uzávěru a otevření dveří, zvonění a znamení řidiči. Ke každému zvuku je také uložen zvuk opačný. Zvuk znamení k řidiči a zvuk signalizace uzávěru dveří je u typu T3R.P totožný.^[2]

V konfiguračním souboru je potřeba použít tečku, nikoliv čárku, jako oddělovač desetinných míst. U všech hodnot kromě nátěrů se bere pouze poslední definovaná hodnota. U nátěrů se základní nátěr neuvádí. Prázdné řádky jsou povoleny. Komentáře jsou řádky které začínají znakem „#“ a jsou přeskočeny. Nepoužité hodnoty jsou nastavené na prázdné řetězce.

Udává-li hodnota informaci spjatou s polohou a má-li číselné hodnoty za sebou, jsou interpretovány jako hodnoty pro osy X, Y, Z v tomto pořadí. Model by měl být nastaven tak, že osa X+ směřuje do levé strany vozidla, osa Y+ na vrch vozidla a osa Z+ čelu vozidla. Má-li hodnota číselné hodnoty za sebou, udává poslední číslice vodorovné otočení po

směru hodinových ručiček ve stupních. Odsazení je vždy udáno od středu článku ve kterém se položka nachází.

Uživatelský název parametru	Název hodnoty	Hodnota
Hmotnost vozidla (v kilogramech)	MASS	17 300
Délka vozidla (v metrech)	LENGTH	15.104
Šířka vozidla (v metrech)	WIDTH	2.5
Výška skříně vozidla (v metrech)	HEIGHT	3.06
Počet sedacích míst ve vozidle	SEATS	22
Maximální počet stojících cestujících	STANDING	69
Počet motorů	MOTORS	4
Výkon motoru (v kilowattech)	POWER	40
Maximální provozní rychlost (v km/h)	SPEED	65
Název podvozků modelu	BOGIENAMES	T3 bogie 1,T3 bogie 2
Přípony náprav podvozků (musí být právě 2)	BOGIEWHEELS	wheels F,wheels B
Seznam meshů přidělených každému podvozku.	BOGIEMESHES	Prázdná hodnota.
Odsazení podvozků, pouze na vodorovných osách X, Z	BOGIEOFFSETS	0.0,3.225;0.0,-3.225
Odsazení podvozků na trati	BOGIETROFFSETS	2.5,8.95
Materiál modelu na který se budou nátěry aplikovat	LIVERYMAT	Material.paint
Nátěry	LIVERY	PLF,soubor textury PID,soubor textury
Odsazení pohledu řidiče	CAMERAOFFSET	0.2,2.3,5.9
Otáčky motoru	MOTORRPM	1750
Hodnota převodovek	GEARRATIO	7.43
Průměr kol metrech	WHEELDIAMETER	0.6
Koeficient aerodynamického odporu	AEROCOEF	1.8

Tabulka 2: Konfigurační hodnoty pro typ T3R.P^{[13][14]} CITACE ENGINEERINGTOOLBOX

Název ovládacího prvku	Název hodnoty	Dodatečná hodnota (TYPE)	Hodnota posunu (OFFSET)	Hodnota rozměru (SIZE)
Přepínač řízení	STARTER		-0.2,1.5,5.9	
Pískovač	SANDER		0.4,1.0,5.9	0.1,0.3,0.3,0.0
Odhlášení zastávky	STOP		TODO	
Zvonek	BELL		TODO	
Dveře řidiče (zde první dveře vozu)	DDOOR		TODO	
Tísňové tlačítko	EMERGENCY		TODO	
Otevření dveří	OPENDOOR	L	TODO	
Uvolnění dveří	FREEDOOR	R	TODO	
Zavření dveří	CLOSEDOOR	M	TODO	
Typ ovládání dveří	DOORS	switch *		
Nouzová brzda	BRAKEE		TODO	0.1,0.2,0.1,0.0
Ruční odbrzdění brzdy (zde čelist'ové/mechanické)	BRAKE1		TODO	
Osvětlení denní	LIGHTDAY	L	TODO	
Osvětlení noční	LIGHTNIGHT	R	TODO	
Typ ovládání osvětlení den/noc	LIGHT	switch *		
Osvětlení, přepnutí režimu	LIGHTMODE		TODO	
Vypnutí osvětlení kabině	CLIGHTON	R	TODO	
Zapnutí osvětlení kabině	CLIGHTOFF	M	TODO	
Typ	CLIGHTTYPE	switch *		

Tabulka 3: Posunové hodnoty pro ovládání vozidla typu T3R.P

* – Buď hodnota *switch*, nebo také *buttons*. Switch označuje přepínač – přepínač má polohy označené *K L M R P*. *K* a *L* jsou na levé straně, *M* je střední poloha a *R*, *P* jsou na pravé straně. Krajní písmena *K*, *P* jsou od středu dále než vnitřní písmena *L* a *R*. Písmen může být nastaveno několik. Pokud je hodnota *buttons*, jsou všechny ovládací prvky pro dané akce od sebe zvlášť.

Základní hodnota pro rozměr ovládacích prvků je *0.1,0.1,0.1,0.0*. (krychle o straně 1dm a obsahu právě 1dm³ bez žádné rotace).

Uživatelský název parametru	Název hodnoty
Soubor modelu s uvedenou příponou .glb	MODEL
Zvuk otevírání dveří	SOUNDOPENDOOR
Zvuk zavření dveří	SOUNDCLOSEDOOR
Zvuk otevírání dveří do kabiny	SOUNDOPENCABIN
Zvuk zavírání dveří do kabiny	SOUNDCLOSECABIN
Zvuk rozjezdu vozidla	SOUNDSTART
Zvuk brzdění	SOUNDBRAKE
Zvuk přejezdu výhybky	SOUNDSWITCH
Zvuk dveřní signalizace a znamení řidiči	SOUNDINFO

Tabulka 4: Další povinné konfigurační hodnoty

2. Implementace vlastní grafické infrastruktury

2.1. Okno grafické rozhraní

2.1.1. Postup implementace

Implementace grafické infrastruktury probíhalo v několika stádiích. Nejdříve byly implementovány základní abstrakce systémových funkcí, následně bylo implementované osvětlení, stíny, načítání 3D modelů, okolí včetně tratí, a fyzické simulace.

2.1.2. Kompatibilita mezi operačními systémy

Pro kompatibilitu mezi různými platformami byl vybrán standardní jazyk C++, konkrétněji verze C++20. Tento jazyk je nejen možné na všech operačních systémech používat, sám je často daných operačních systémech použit.^[15]

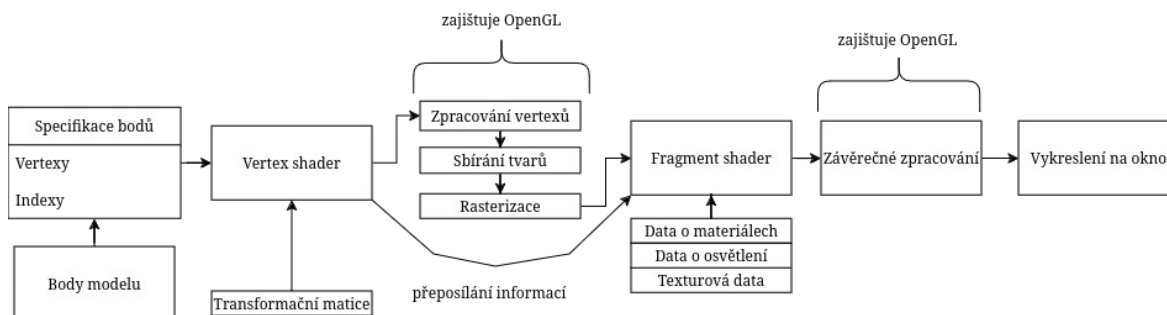
Z téhož důvodu bylo vybráno API OpenGL, které je podporované na všech nejrozšířenějších operačních systémech – Windows, MacOS i Linux. Na něj navazují všechny ostatní technologie při vývoji použité, jako třeba GLFW („OpenGL Framework“), GLEW („OpenGL Extension Wrangler“) a GLM („OpenGL Mathematics“).

Kompatibilita systému vypisování barevného textu byla docílena rozdělením funkce přebarvení na 2 části – část pouze pro GNU/Linux a část pouze pro Windows. Pro část GNU/Linux byly využity tzv. ANSI únikové kódy^[16].

Současně kompilační infrastruktura programu, založena na systému Cmake, podporuje pouze operační systémy Windows (přes překladač MinGW) a Linux (přes překladače GCC nebo Clang). Podpora platformy MacOS by po menších změnách měla být možná.

Editory pro hlásič, scénář a linku jsou napsány jako webové stránky v JavaScriptu. Toto znamená že je možné editory spustit na jakémkoliv zařízení s monitorem. Na mobilních zařízeních je sice spuštění možné, není to avšak oficiálně podporované a povede to k degradovanému uživatelskému prostředí.

2.1.3. Infrastruktura vykreslování objektů na obrazovku



Obrázek 6: Proces vykreslování modelu grafickým rozhraním OpenGL 4.5

OpenGL používá jistý proces (viz obrázek 6) při vykreslování modelů.^[17]

Jako první čte vstupní data nahrané programem do paměti grafické karty počítače – vertexy (body) a indexy (pořadí bodů). Dále jsou nastaveny 2 shadery – vertexový shader a fragmentový (či pixelový) shader. Vertexový shader je spuštěn pro každý bod na modelu. Fragmentový je spuštěn jednou pro každý pixel na modelu. Z vertexového do fragmentového shaderu je možné přeposílat informaci uvedenou v bodech. Program může také se shadery komunikovat za pomoci SSBO a UBO. Přes SSBO je možné posílat pouze skalární hodnoty, 2-rozměrné vektory 4-rozměrné vektory. 3-rozměrné vektory z důvodu paměťové struktury OpenGL posílat nelze¹, z téhož důvodu je potřeba vždy nastavit 16-bitové zarovnání paměti u posílaných objektů.^[18*]

Po načtení dat je spuštěn nejdříve shader vertexový, následně pak fragmentový. Ve vertexovém je určena pozice bodu a přeposílána informace do shaderu fragmentového.

[17*][19]

Ve fragmentovém shaderu jsou implementovány veškeré výpočty týkající se osvětlení a barev. Ve vertexovém shaderu jsou implementovány veškeré posuny a změny vertexů či meshů. Geometrové shadery (angl. Geometry shaders), které mohou přidávat nové vertexy, nejsou v simulátoru použity.^[17*]

V rámci optimalizací a vylepšení kvality vykreslování byly použity funkce API OpenGL „face culling“, neboli nevykreslování zadních stran stěn, a s tím související funkce „depth testing“ (hloubkové testování), která vykresluje pouze stěny které jsou z pohledu uživatele vidět.^[20 p.174-181] Pro vylepšení kvality byla použita funkce „anti-aliasing“ (konkrétněji technologie MSAA), jenž funguje na základě průměrování sousedních pixelů a tím zhlazuje strmý přechod mezi jednotlivými pixely.^[20 p. 264-271]

1 Dle standardy OpenGL je možné posílat 3-rozměrné vektory, žádné implementace to jednak nepodporují často to vede neřešitelným problémům.^[18]

2.1.4. Grafické rozhraní

Grafické rozhraní bylo implementováno za pomoci knihovny Dear ImGui. Použití grafického rozhraní (GUI) v simulátoru je možné rozdělit na části: hlavní menu, simulační a na pauzovací.

Při spuštění programu se objeví hlavní menu. Z hlavního menu je možné začít jízdu simulátoru, změnit nastavení simulátoru či ho také opustit. Menu nastavení obsahuje jednotlivé možnosti pro změnu parametrů simulace či změnu mapy, linky či hlásiče. Změna vozidla či nátěrů není zatím v nastaveních podporována a musí se docílit přes konfigurační soubor.

Během režimu simulátoru je dostupné simulační rozhraní. Rozhraní nese většinou informativní charakter – informuje o stavu scénáře, o počtu nasbíraných bodů během jízdy, o současné rychlosti tramvaje a o jiných potřebných během jízdy informacích. Je také přítomna funkce vypínání a zapínání grafického rozhraní za pomoci klávesy *F1*.

Z menu pauzy je možné simulátor opustit, vrátit se do hlavního menu či změnit nastavení simulátoru.

2.2. Vykreslování modelů

2.2.1. Načítání vykreslování modelů, systém GMS

Modely jsou obvykle rozdělené na několik částí, odborně nazývaných meshe. Toto rozdělení je provedeno autorem. Simulátor podporuje pouze formát GLTF, který je založen na formátu JSON. Nejdříve jsou načteny všechny materiály modelu, případně textury daného materiálu. Dále je načtena hierarchie bodů a jejich transformace. Nakonec program načítá jednotlivé meshe modelu. Pro každý mesh jsou načteny všechny jeho vertexy, jejich normálové vektory, texturové souřadnice určující jakou část textury má daný vertex použít, indexy a následně jiné parametry vertexu, například indexy kostí jejich koeficienty vlivů.

Všechny materiály jsou uloženy globálním úložišti, označovaném programem jako „Global Material Store“, zkráceně GMS. Samotné meshe ukládají pouze odkaz na danou položku GMS. Materiály ukládají svoje parametry pro model světla svojí texturu, pokud jí nemají pak svojí barvu. Za pomoci systému GMS je zajištěna možnost změny materiálu meshe – každý materiál má uložené také jeho varianty. Pro jednodušší identifikaci „variant“ materiálu má každý materiál také identifikační text – pro standardní variantu je použit speciální, předem definovaný text. Varianty materiálu jsou přidány uživatelem, program automaticky načítá pouze standardní variantu. Systém GMS také eliminuje duplikaci totožných materiálů. Totožné materiály jsou detekovány dle souborové cesty textury. Během práce programu není možné měnit materiály meshe, je možné avšak využít variantového systému GMS. Každý materiál může mít přiřazené varianty. Tento systém změn variantů je simulátoru užíván například u nátěrů vozů, u světel v domech či u ročních období.

Vykreslování modelů probíhá v několika stupních. Vertexy a indexy jsou automaticky poslány do fragmentového a vertexového shaderu. Parametry materiálů jsou uloženy systémem GMS do SSBO, který následně informaci přepośle do fragmentového shaderu, který pak pro každý pixel vypočítává barvy a aplikuje dodatečné efekty, například světla nebo stíny.

2.2.2. Načítání vykreslování textur modelů

Během načítání souboru modelu se taky načítají textury jednotlivých modelů. Každý materiál má přiřazenou právě 1 texturu a každý vertex meshe má přiřazen právě jeden materiál. Celkem ve stejný čas je možné v jednom fragmentovém shaderu využít právě 32 textur², některé těchto texturových pozicí jsou ale rezervovány pro použití systémem stínů, zaujímající současně 1 slot pro každé světlo které má simulované stíny, tedy celkem 4 sloty, nebo systém počasí který současně zabírá 1 slot pro detekci střech. Pozice pro textury systému vykreslování stínů se vždy počítají od konce. Materiály nemusí mít pouze texturu – mohou mít také nastavenou barvu při vykreslování mohou použít oboje. Tato hodnota je vyjádřena parametrem materiálu $textureAmount \in (0;1)$, který určuje procentuální podíl textury na výslednou barvu materiálu. Je-li tato hodnota rovna nule, neboli nemá-li materiál žádnou přiřazenou texturu, nezabírá v shaderu žádný texturový slot. Z tohoto plyne zásadní limitace simulátoru – každý mesh může obsahovat nanejvýše 27 texturovaných materiálů.

2.2.3. Vykreslování průhledných částí modelů

Vykreslování průhlednosti není zcela jednoznačnou věcí. Pro vykreslování všech průhledných částí za sebou, a to nezávisle od pořadí, je potřeba využít technologii OIT, celým názvem „order independent transparency“ neboli česky „průhlednost nezávislá na pořadí“.^[21*]

Existuje několik technologií OIT – simulátor používá technologii „weighted-blended“, vynalezenou v roce 2013 společností NVIDIA. Technologie funguje na principu vykreslování průhledných a neprůhledných objektů zvlášť, a jejich následné kombinování před vykreslením. Pro podporu anti-aliasingu jsou avšak v simulátoru implementovány některé změny.^[21*]

Framebuffery, na které se průhledné a neprůhledné části vykreslují, jsou nastaveny na tzv. „multisamplované“, ve kterých každý pixel může mít několik hodnot.^[20 p.268] Toto pomáhá plynulejším přechodům mezi hranicemi objektů. Tato změna zvyšující kvalitu obrazu avšak požaduje přidání dalšího kroku, a to překopírování již kombinovaného multisamplovaného bufferu na buffer nemultisamplovaný, který je následně vykreslen na obrazovku.

Neprůhledné objekty jsou vykreslovány do jejich framebufferů beze žádných změn. U průhledných objektů se provádí dodatečné výpočty a ukládají se 2 hodnoty – upravená zprůměrovaná barva a výsledná průhlednost daného bodu na obrazovce.

2 Většina moderních grafických procesorů podporuje 32 slotů či více, samotné API OpenGL avšak určuje minimální počet textur dostupných ve fragmentovém shaderu jako 16.

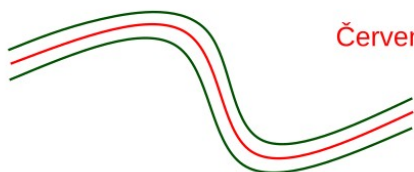
2.3. Vykreslování okolí

2.3.1. Vykreslování kolejí

Generace vertexových bodů tratí v simulátoru

Šířka tratě, neboli rozestup mezi posunutými křivkami, je možné měnit.

Tratě definovány jako Bézierovy

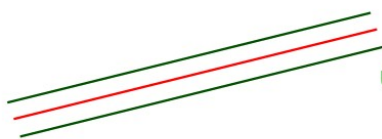


Červeně je vyznačena základní trať jak je ve scénáři definována.

Zeleně jsou vyznačeny posunuté tratě. Každá zelená křivka označuje 2 posunuté křivky dle kterých jsou body definovány.

Původní Bézierova křivka je kubická - definována 4 body: počátečním, konečným a 2 středními. Z každé křivky je vzat jeden bod. Ty jsou následně indexovány. Výška každého bodu je vypočítána z lineární interpolace dvou nejbližších výškových bodů.

Tratě definovány jako rovné



Červeně je vyznačena úsečka mezi 2 koncovými body.

Zeleně jsou vyznačeny posunutí úsečky. Každá zelená úsečka označuje 2 posunuté úsečky dle kterých jsou body definovány.

Z každé křivky je vzat jeden bod. Ty jsou následně indexovány. Pozice každého bodu je rovna lineární interpolaci koncových bodů s určitým poměrem. Rovná trať obsahuje stejný počet bodů jako je počet výškových bodů. Výška tratě v jednotlivém bodě je následně rovna výšce výškového bodu.

Obrázek 7: Obrázek popisující generaci tratí

Koleje jsou generovány za pomoci kubických Beziérových křivek v dvourozměrném prostoru. Převýšení tratě se určuje za pomoci 10 tzv. výškových bodů uložených společně s body potřebnými pro vygenerování křivky. Opravdová výšková hodnota bodů je u tratí definovaných Bézierovými křivkami vypočtena za pomoci lineárních interpolací výškových bodů.

Veškeré koleje jsou uloženy v jednom páru vertexového a indexového bufferu. Tato struktura má několik výhod – za prvé je použit pouze 1 kreslicí příkaz, což drasticky zrychluje vykreslování, a za druhé jsou vždy vidět všechny koleje, což značně pomáhá pocitu realismu.

2.3.2. Vykreslování silnic, země, ulic, kolejových svršků stěn

Pro texturové parcely, které jsou použity pro terén, povrch ulic, kolejové svršky i pro stěny, jsou při načítání scénáře generované vertexové a indexové buffery. Veškeré texturové parcely jsou zkombinované do jednoho vertexového a jednoho indexového bufferu, jehož body odkazují na jednotlivé materiály (a s nimi spojené textury) v systému GMS. Tento přístup, stejně jako tratí, má výhodu že je terén vždy vidět a není ho potřeba při změně zastávky znovu načítat či jinak měnit. Tento přístup má avšak také jednu nevýhodu, a to ohrazení počtu různých typů texturových parcel na 27 z důvodů ohrazení počtu texturových slotů API OpenGL (viz kapitola 2.2.2.).

Jednotlivé textury byly staženy z webového portálu AmbientCG.^[22] Materiál na svrších či okolních silnicích nemá žádný vliv na fyzikální simulaci či pohyb vozidla.

2.3.3. Vykreslování budov památek
















Vykreslování budov funguje za pomoci využití instancovaného vykreslování. Pro každý typ budov, kterých je v simulátoru 4, je poslán grafické kartě pouze jeden příkaz na vykreslování objektů.

Jelikož památky jsou jedinečné, a většinou jsou přítomny na mapě pouze jednou, není vykreslování památek instancováno, a z toho důvodu že instancování by nepřineslo žádné podstatné zrychlení programu. V tomto případě by mohlo vést i k značnému zpomalení.

3. Prvky předpisy ovládání simulátoru

3.1. Pohyb po trati návěstí

Seznam tramvajových návěstí, jejich funkcí a vzhledu
 Návěsti jsou značky které určují co má řidič tramvaje dělat resp. nedělat.

<p>Základní návěsti</p> <div style="display: flex; align-items: center; margin-bottom: 5px;">  <div style="margin-left: 5px;">Č. 1 označuje zajišťovací výhybky. Takovou je možné najít např. na obratišti Zvonařka.</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;">  <div style="margin-left: 5px;">Č. 2 označuje návěst stůj. Vlak nesmí návěst projet. Používá se při výlukách.</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;">  <div style="margin-left: 5px;">Č. 3 označuje přítomnost přejezdu s předností tramvaje.</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;">  <div style="margin-left: 5px;">Č. 4 označuje maximální rychlost úseku je-li nižší než očekávána t.j. stanovena jinými předpisy.</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;">  <div style="margin-left: 5px;">Č. 5 označuje konec úseku označeného značkou č. 4.</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;">  <div style="margin-left: 5px;">Č. 6 označuje návěst "pomalu" neboli omezení rychlosti na 10 kilometrů za hodinu.</div> </div> <p>Rychlostní návěsti</p> <div style="display: flex; align-items: center;">  <div style="margin-left: 5px;">Č. 7 označuje úsek s maximální rychlostí.</div> </div>	<div style="display: flex; align-items: center; margin-bottom: 5px;">  <div style="margin-left: 5px;">Č. 8 označuje konec úseku označeného značkou č. 7.</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;">  <div style="margin-left: 5px;">Č. 9 označuje místo které tramvaj musí projet nejvýše uvedenou rychlostí.</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;">  <div style="margin-left: 5px;">Č. 10 stanovuje rychlostní limit 10 km/h při odbočování z důvodu struktury výhybky.</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;">  <div style="margin-left: 5px;">Č. 11 označuje oblouk o malém poloměru $r < 25m$ a maximální rychlost 15 km/h na něm.</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <div style="display: flex; gap: 5px;"> <div style="border: 1px solid black; padding: 2px;">ZONA 22-6 h 30</div> <div style="border: 1px solid black; padding: 2px;">ZONA 22-6 h 40</div> </div> <div style="margin-left: 5px;">Značky č. 12 a 13 určují maximální rychlost v určitý čas.</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <div style="display: flex; gap: 5px;"> <div style="border: 1px solid black; padding: 2px;">ZONA 22-11 h 30</div> <div style="border: 1px solid black; padding: 2px;">ZONA 22-11 h 40</div> </div> <div style="margin-left: 5px;">Značky č. 14 a 15 označují konec úseku určeného značkami 12 a 13.</div> </div>	<p>Návěsti pro sběrač</p> <div style="display: flex; align-items: center; margin-bottom: 5px;">  <div style="margin-left: 5px;">Č. 16 - Úsekový dělič. Tramvaj musí projet toto místo bez odběru proudu.</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;">  <div style="margin-left: 5px;">Č. 17 - Úsekový dělič s magnety. Je možné toto místo projet s odběrem.</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;">  <div style="margin-left: 5px;">Č. 18 - Stáhni sběrač. Označuje překážku v trolejovém vedení.</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;">  <div style="margin-left: 5px;">Č. 19 - Zvedni sběrač. Po této návěsti je plně bezpečné zvednout sběrač.</div> </div>
--	--	---

Obrázek 8: Funkce, čísla vzhled jednotlivých tramvajových návěstí.^{[1][23][24]}

Tramvaj se pohybuje po kolejích. Povolená rychlost a povolené či povinné akce tramvaje jsou dány návěstmi a polohami zastávek.

Na obrázku nahoře je dán seznam nejčastěji užívaných návěstí. Na rovných úsecích (tj. Na úsecích bez výhybek) je rychlost dána návěstmi č. 4 až 15. Návěsti se mohou také týkat polohy sběrače – zda ho musí řidič tramvaje stáhnout či naopak zvednout. Tímto se zabývají návěsti č. 16 až 19.^[1 *]

Při uslyšení zvukové sirény vozidla IZS, musí řidič tramvaje neprodleně zastavit.^[1]

3.2. Výhybky

3.2.1. Elektrické

Elektrické výhybky jsou ovládány uzavřením elektrického obvodu výhybky za pomoci pantografu vozidla.

Elektrické výhybky je ke 22. březnu 2025 možné v pražské tramvajové síti potkat pouze na 2 místech, to na vjezdu do smyčky Dlabačov ze směru Pohořelce na lince 22/23 a na vjezdu do smyčky Vysočanská. Do roku 2025 byla síti přítomna ještě jedna elektricky ovládaná výhybka, a to nedaleko zastávky Chotkovy sady, ta byla avšak během výluky přestavěna na rádiově ovládanou.^[1 *]

3.2.2. Rádiové

Nejčastější typ výhybek na pražských tratích jsou rádiové. Rádiově ovládané výhybky mají rádiový přijímač, který čte informaci o přehození výhybky poskytnutou tramvají. Dle přijetího signálu se pak přehodí do odpovídajícího směru.^[1 *]

3.2.3. Zajišťovací výhybky

Zajišťovací výhybky, též označované jako vratné, jsou takové, které se po průjezdu vozu za pomoci pružiny vrátí na původní pozici. Takové výhybky se současně používají pouze na obratišti Zvonařka pro linku 23 a ve vozovnách. Z důvodu větší opatrnosti potřebné při manipulaci s těmito výhybkami, jsou označeny zvláštní návěstí která je na seznamu návěstí (obrázek 8) označena číslem 1.^[1 *]

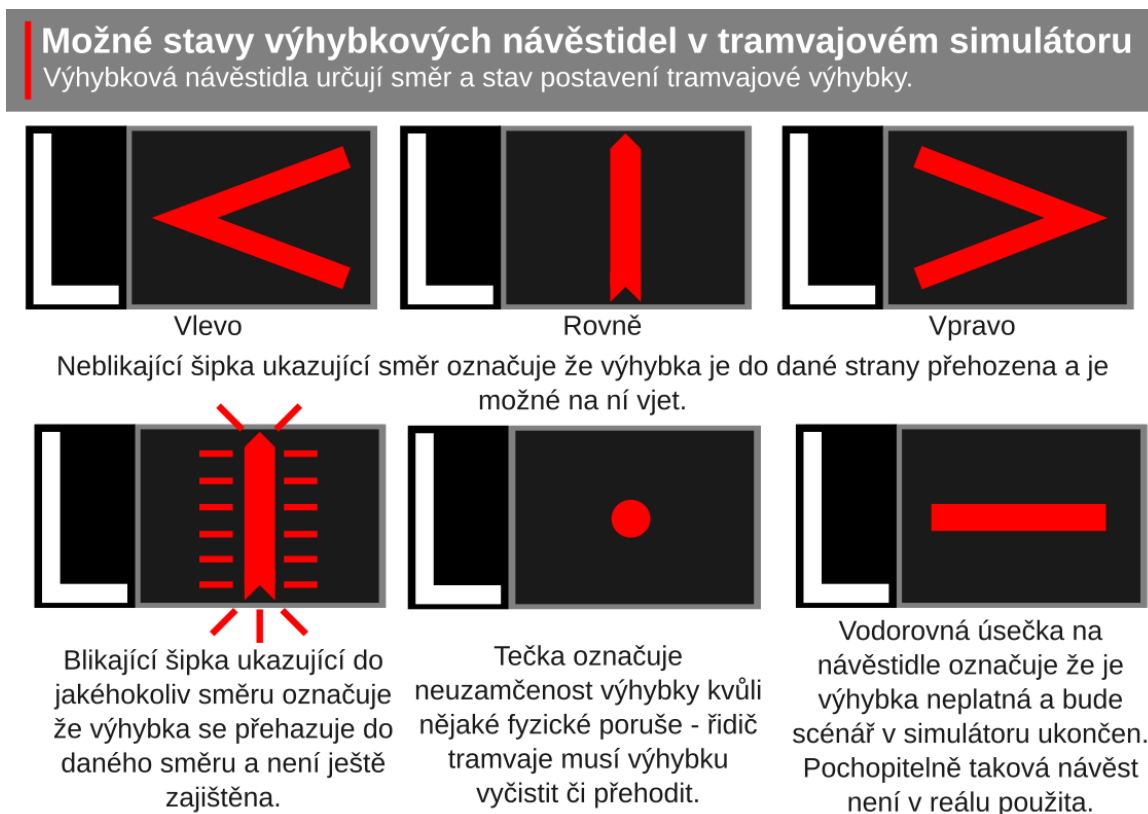
3.2.4. Přehazování výhybek

Přehazování výhybek závisí na typu výhybky. Elektrické výhybky jsou přehazovány trolejovým kontaktem. Rádiové výhybky jsou přehazovány rádiovým kontaktem.^{[1 *][2 *]}

Při poruše přehazování výhybky je zapotřebí přehodit danou výhybku ručně. Toto se dělá za pomoci tzv. „stavějícího klíče“, přezdívaného jako „tágo“.^{[1 *][2 *]} Klíč se umístí do správného místa a následně se výhybka může přestavit do směru právě řidiči potřebného.

3.3. Návěstidla

3.3.1. Výhybková návěstidla



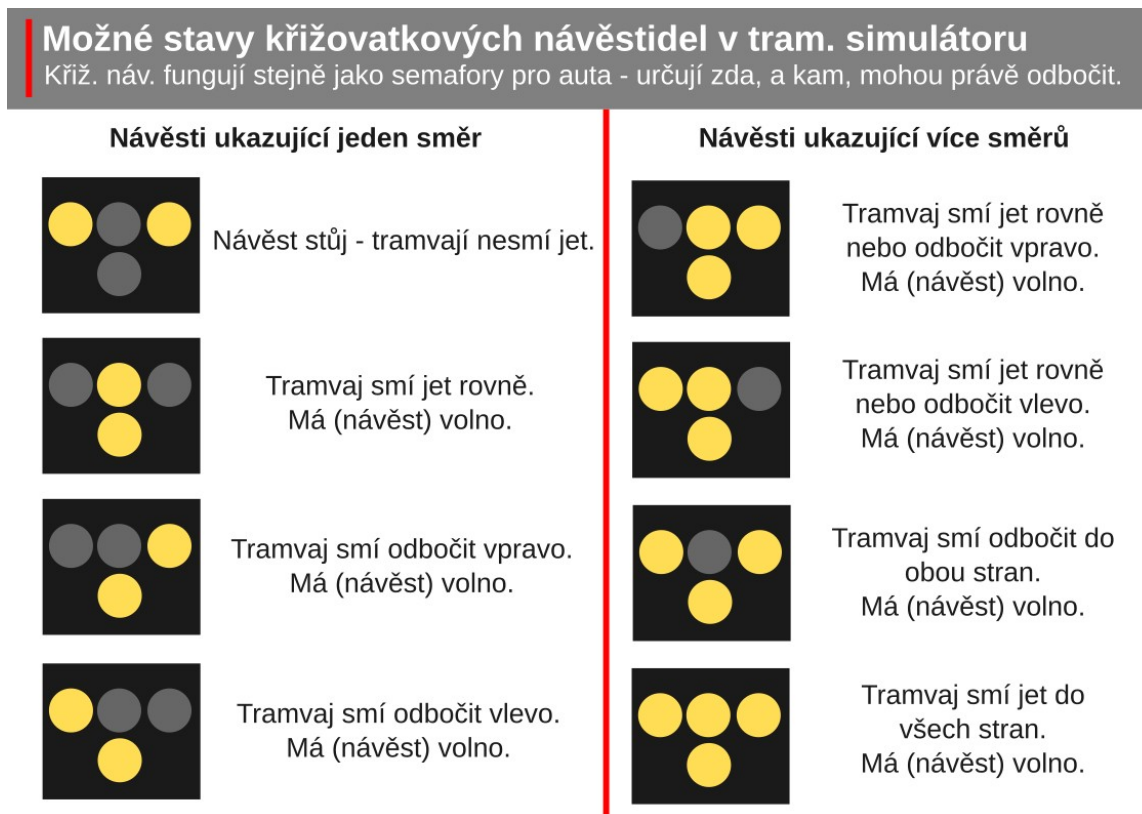
Obrázek 9: Možné stavy výhybkových návěstidel tramvajovém simulátoru.^[1]

Návěstidla určují směr a stav přehození výhybky. Na samotném displeji návěstidla je znázorněna červená šipka znázorňující směr do kterého je současně výhybka přehozena. Blikající návěst označuje že výhybka není ještě zajištěna je pouze v procesu přehazování.

[1*]

Tramvaj nesmí opustit sféru vlivu návěstidla dokud není výhybka přehozena a zajištěna. Vyjede-li tramvaje ze sféry vlivu návěstidla, scénář se ihned ukončí. Při normální jízdě by toto znamenalo rozbití výhybky a následná vnitropodniková pokuta pro způsobilého řidiče.

3.3.2. Křižovatková návěstidla



Obrázek 10: Možné stavy křižovatkových návěstidel tramvajovém simulátoru.^{[1][25]}

Křižovatková návěstidla plní funkci semaforu pro tramvaje. Jednotlivé funkce křižovatkových návěstidel jsou obsaženy na obrázku.

Simulátor současně nesimuluje výzvoová návěstidla, a to z důvodů jejich nečetnosti a jejich nevelkého vlivu na silniční provoz. Na rozdíl od „normálních“ křižovatkových návěstidel, též přezdívaných jako „čočky“, která jsou kontrolována odpovědnými drážními předpisy, jsou návěstidla výzvoová kontrolována přímo Dopravním podnikem hl.m. Prahy, a tak jsou používány pouze na místech kde se na křižovatce mívají pouze tramvaje a není přítomen žádný silniční provoz. I tak jsou na takových křižovatkách přítomny i standardní křižovatková návěstidla.^[1*]

3.3.3. Předvěsti křižovatkových návěstidel

Křižovatková návěstidla mohou mít také předvěsti, která ukazují řidiči stav následujícího návěstidla. Od obvyklých křižovatkových návěstí se vizuálně liší pouze žlutou tabulkou s černým písmenem “P”. V simulátoru se od obvyklého také liší tím, že nemá přiřazenou žádnou sféru vlivu či žádnou funkčnost – pouze informuje stavu následujícího návěstidla.^[25*]

3.4. Jízdní řád

3.4.1. Normální zastávky

Na normálních zastávkách je vůz povinen zastavit. Není-li zastávka kontrolním bodem, není dán přesný čas kolik musí řidič na zastávce strávit. Není-li na zastávce žádný cestující přítomen, nemá řidič za povinnost otevírat dveře.^{[1 *][2 *]}

3.4.2 Zastávky na znamení

Na zastávkách na znamení nemusí řidič vždy povinně zastavit. Je-li na zastávce cestující, zastavit je avšak povinen. Nemá-li řidič jistotu přítomnosti cestujících na zastávce, musí taktéž zastavit.^{[1 *][2 *]}

3.4.3 Kontrolní body

Stejně jak je to reálných jízdních řádů, řidiči musí dodržovat časy pouze pro některé zastávky, tzv. kontrolní body. Kontrolní body jsou definovány pro každou linku zvlášť. U kontrolních bodů není možné odjet předčasně.^{[1 *][2 *]}

Ve vnitropodnikových jízdních řádech vozidla jsou specifikovány pouze časy kontrolních bodů, nikoliv všech zastávek.^{[1 *][2 *]}

3.4.4. Konečné zastávky

Na konečných zastávkách končí linky svou jízdu. Toto však neznamená že řidič končí směnu nebo že řidič vždy končí směnu na konečné.^[1 *]

V simulátoru nejsou přidány žádná ohraničení na konečné zastávky. Toto řešení povoluje simulovat části linek či linky výlukové, které nemusí povinně končit na obratišti, nýbrž taktéž na úvratí či, povoluje-li to typ tramvaje, uprostřed tratě.^{[1 *][2 *]}

3.5. Identifikace tramvaje

Tramvaj je identifikována dvěma hodnotami – číslem vozovny, a pořadím výjezdu z vozovny. Jejich přesný časový rozsah není definován, stejně jako není žádné jednoznačné pravidlo pro jejich tvorbu. To také znamená, že pořadí není určeno dle času odjezdu.^[26*]

Tyto hodnoty jsou závislosti na typu vozu tramvaje zobrazeny na různých místech. Novější tramvaje typu KT8D5.RN2P, 14T, 15T a rekonstruované vozy T3R.PLF mají pořadí zobrazené digitálně a (kromě T3R.PLF) na straně vozu. Vozy typů T3R.P a nerekonstruované T3R.PLF mají své pořadí uvedené na čelní straně vozu kabině řidiče a to ve formě plastové destičky.^[1]

Název vozovny	Kód vozovny
Vozovna Hloubětín	1
Vozovna Kobylisy	2
Vozovna Motol	3
Vozovna Pankrác	4
Vozovna Strašnice	5
Vozovna Střešovice	6
Vozovna Vokovice	7
Vozovna Žižkov	8

Tabulka 5: První číslice dle vozovny^{[1][26]}

První číslice pořadového čísla určuje vozovnu. Přiřazení čísel jednotlivým vozovnám je uvedeno v tabulce. Druhá a třetí číslice určuje pořadí ve vozovně. Toto číslo závisí na jednotlivé vozovně a nikoliv na času odjezdu není ho tím pádem možné přesně simulovat.

Pořadové číslo spoje není to stejné jako pořadí na lince. Pořadí na lince, které je dostupné cestujícím on-line, označuje pořadí spoje v rámci určité linky. Obvykle je uvedeno ve formátu *LL/P*, kde *L* označuje linku a *P* označuje pořadí vlaku na lince. Pořadové číslo není na veřejných zdrojích uvedeno. Pořadové číslo linky se čte následovně:

P-tá LL-ka: např. devátá šestnáctka

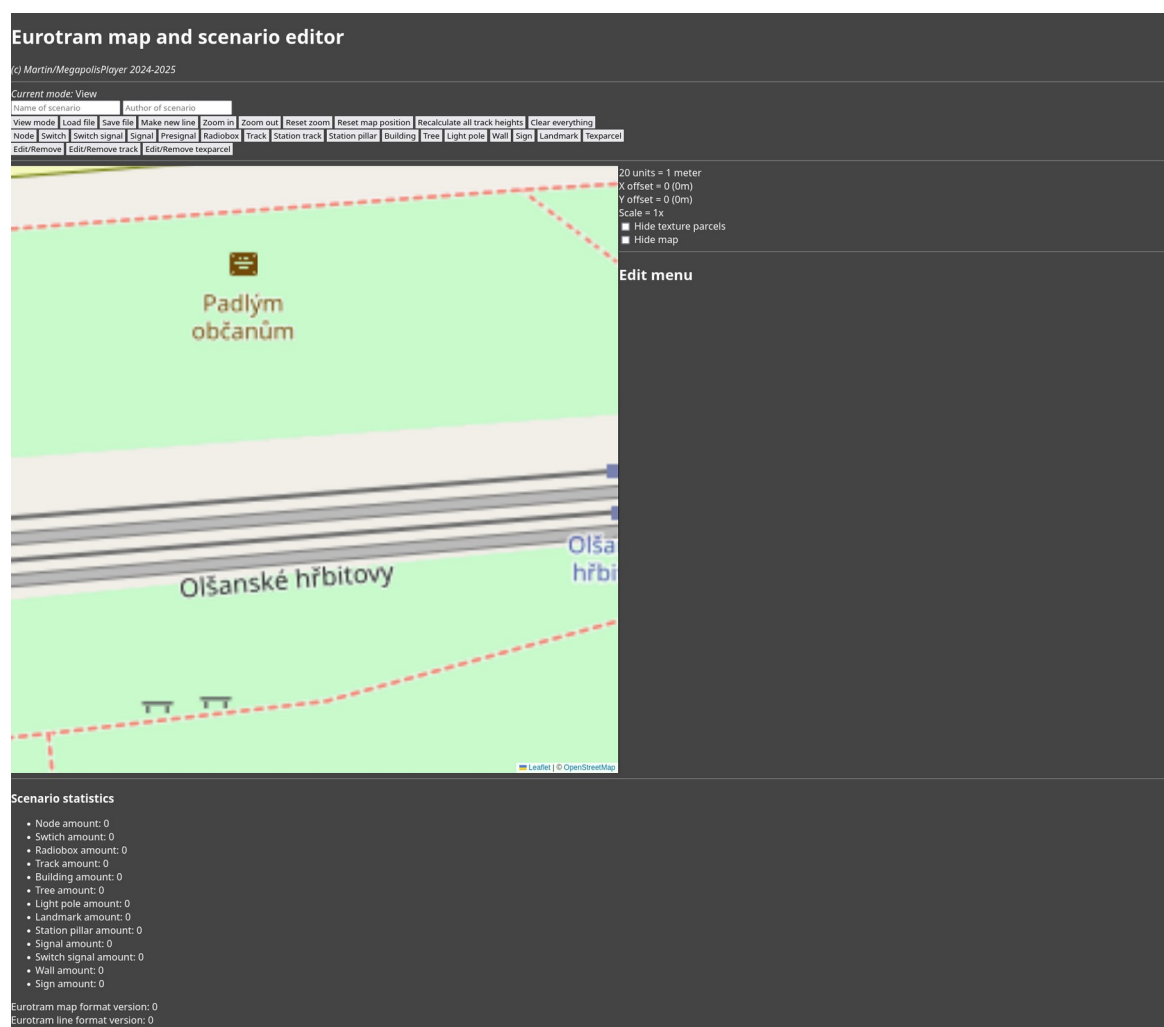
[1*][26*]

4. Dodatečné nástroje formáty jejich souborů

4.1. Nástroj na úpravu mapy a scénářů

4.1.1. Základní informace

Nástroj na úpravy (jinak označovaný jako „editor“ map a scénářů) je webová aplikace napsaná v programovacím jazyce JavaScript (za pomoci dodatečných technologií HTML a CSS).^[27] Funguje jako zcela nezávislý program který exportuje „mapu“ (kolekci tratí, zastávek okolí) do vlastního formátu. Exportuje také „scénáře“, neboli soubory ve kterých je uložena linka (včetně případného přejezdu na jinou či zatahování do depa) a její jízdní řád.



Obrázek 11: Fotografie nástroje pro úpravu mapy scénářů při jeho otevření.

Editor je rozdělen na 4 části – horní lištu operací, menu úprav, plátno a statistiku.

Statistika pouze ukazuje informaci o scénáři. Ukazuje verzi formátu souboru kterou editor vytváří a počet prvků ve všech seznamech objektů scénáře.

Úpravy probíhají na mapovém podkladu, vytvořeným za pomoci technologie Leaflet.js^[28] používající mapu OpenStreetMap.

Ukládání probíhá ve dvou stupních. Nejdříve je veškerá informace uložena do pole bytů. Toto pole bytů je následně převedeno na soubor a staženo. Načítání funguje opačně – nejdříve se data načtou ze souboru do pole bytu a následně se byty přečtou a interpretují.

Editor sice desetinná čísla nezakazuje, snaží se je ale omezit. Při ukládání do souboru se všechny hodnoty převedou na jejich dolní celá čísla.

4.1.2. Horní lišta

Horní lišta operací ukazuje seznam akcí které může uživatel provést. Z této lišty je možné přidat body, výhybky, návěstidla, rádiové čtečky, tratě, staniční tratě, stromy, pouliční lampy, budovy, památky, stěny a parcely.

Body určují kudy trať prochází. Znájí svojí polohu, výšku a sounáležitost ke stanici. Bod může také hraničit na pomezí stanic – takovém případě je při načtení souboru simulátorem bod duplikován.

Výhybky plní stejnou funkci jako body. Znájí avšak kromě toho také po jakém bodě následují, body které s každé strany do ní směřují, připojenou rádiovou čtečku a návěstidlo. Výhybky nemohou hraničit s jinou stanicí.

V programu je možné najít typy návěstidel – výhybková a křižovatková. Výhybkové návěstidlo ukazuje stav směr výhybky. Křižovatkové návěstidlo funguje jako alternativa semaforu. Fungování návěstidel je podrobněji popsáno kapitole 3.3 „Návěstidla“.

Rádiové čtečky čtou a ukládají požadavek tramvaje, který je pak přečten výhybkou která se následně odpovědně přehodí. Stav výhybky je následně vypsán na výhybkové návěstidlo.

Tratě jsou uloženy jako Beziérovky křivky nebo jako přímky mezi body. Obsahují také 10 rovnoměrně rozložených „výškových bodů“ určujících změnu převýšení. Existuje také tzv. „staniční trať“, která určuje místo zastavení tramvaje. Staniční tratě i normální tratě jsou uloženy ve stejném poli. Tratě nemohou vést mezi dvěma výhybkami.

Stromy ukládají pouze svojí polohu a výšku. Pouliční lampy ukládají také svůj úhel ve stupních.

Budovy ukládají kromě své polohy výšky také svůj typ. Rozměr budovy není ukládán jelikož je dán typem budovy. Jak v nástroji na plánování, tak i v simulátoru je uložena tabulka s rozměry různých typů budov.

Památky jsou zvláštní typ budov. Ukládají pouze polohu, výšku a 4-písmenný kód, který je pak v simulátoru propojen s modelem dané památky či neobvyklé budovy.

Stěny ukládají pouze svoje body, svojí výšku a název svého materiálu.

V každém objektu (kromě památek) je také uložen kód stanice (resp. Staničního okolí) které je objekt součástí. Kód stanice je uložen jako 4-písmenný z kód důvodů optimalizace. V simulátoru je tento kód uložen jako 32-bitové číslo. Z téhož důvodu je podporována pouze písmenná sada UTF-8, ve které je každé písmeno uloženo jako právě 1 byte.

Pro modelování silnic a uličního povrchu je možné přidat tzv. „parcely“, které vymezují určitý prostor v určité výšce. Každá parcela je čtyřúhelník který kromě polohy, výšky a všech svých bodů zná také název svého materiálu. Simulátor pak dle uloženého názvu najde odpovědný materiál v systému GMS a vykreslí z jeho pomoci danou parcelu.

4.1.3. Menu úprav

Menu úprav je umístěno pravé strany od plátna. V tomto rozhraní je možné měnit přiblížení plátna, nastavovat některé parametry vykreslování plátna a upravovat jednotlivé objekty. Jsou dostupné 2 parametry vykreslování plátna, a to vypnutí vykreslování parcel či vypnutí mapového podkladu.

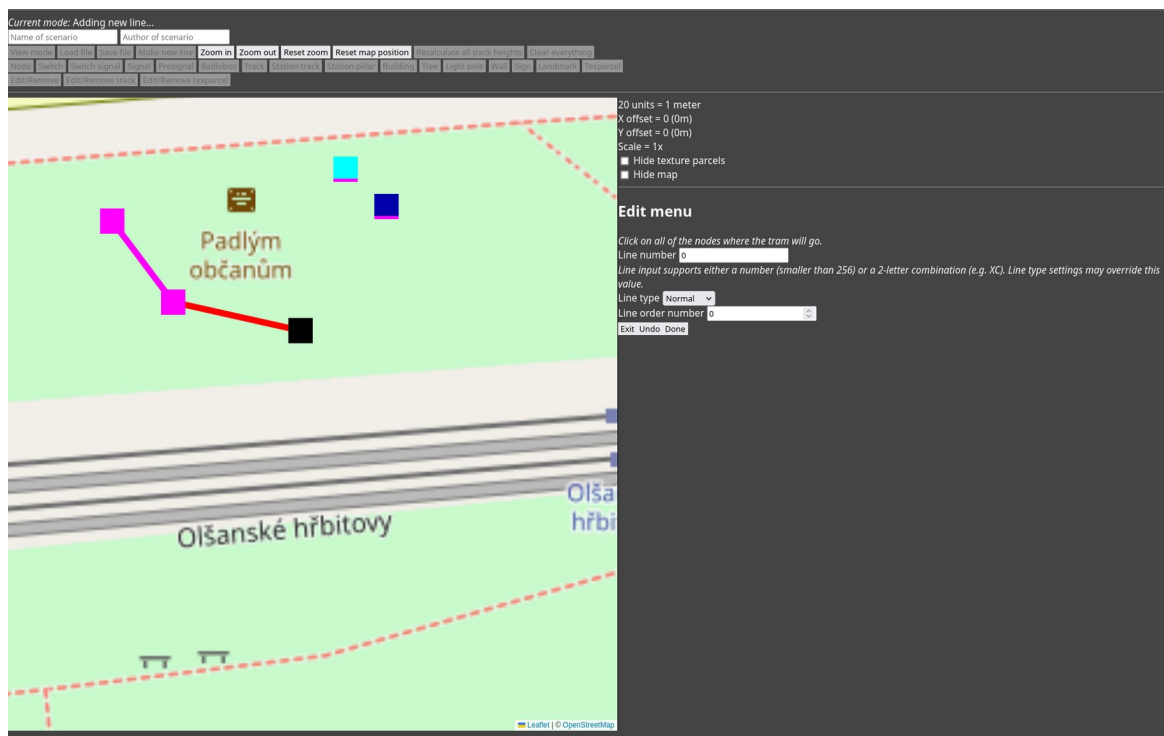
Menu úprav je dynamicky generováno pro každý objekt. Před generací je menu resetováno.

4.1.4. Plátno úprav

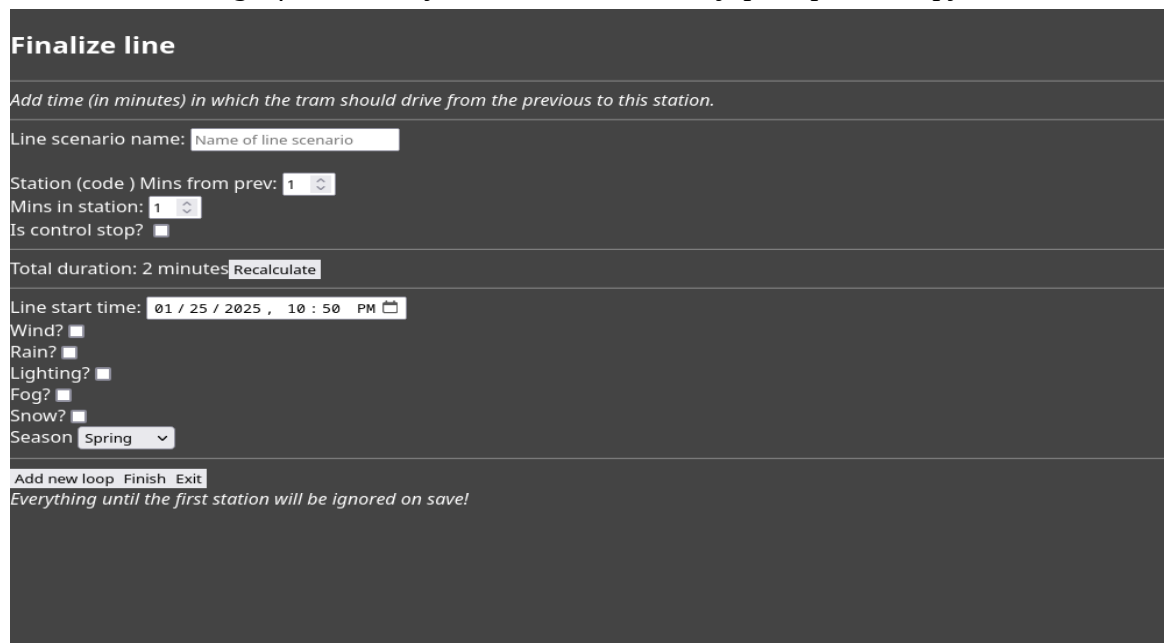
Plátno editoru je možné zmenšovat, zvětšovat a pohybovat s ním. Toto společně s vykreslováním objektů na jejich správných místech je docíleno použitím transformačních matic. Při vykreslování objektu je nejdříve aplikována transformace do centra objektu. Následně, kdy centrum objektu je počátkem soustavy, je aplikována rotace kdy je matice otočena o odpovídající počet stupňů, uvedený objektu. Následně je aplikována translace rovna polovině rozměru objektu, a to z důvodu přenesení počátku na levý horní roh objektu, což všechny kreslicí funkce v jazyce JavaScript očekávají.^[27]

Objekty, které jsou za hranicí plátna nejsou vykreslovány. Samotné plátno je také naprosto průhledné, jelikož obsahuje také mapový podklad mapy OSM vykreslený za pomoci technologie Leaflet.js. Pod mapou a plátnem je také uložené bílé pozadí, které je vidět při vypnutí mapy. Mapu je možné během upravování vypnout. Na základní mapový podklad vytvořený technologií Leaflet.js bylo napojeno vlastní ovládání, které pohybuje nejen mapou ale také plátnem.^[28 *]

4.1.5. Tvorba linky



Obrázek 12: Fotografie režimu vytváření linek v nástroji pro úpravu mapy scénářů.



Obrázek 13: Fotografie menu nastavení linek v nástroji pro úpravu mapy scénářů.

Tvorba linky a s ní spojeného souboru scénáře probíhá po stisknutí tlačítka pro vytvoření nové linky. V tomto režimu se nejdříve vybírá trasa prvního kola linky klikáním na body výhybky, po čemž se nástroj přepne do režimu nastavení linky. V tomto menu může uživatel buď linku ukončit, po čemž se všechna kola linky uloží do souboru, nebo přidat nové kolo linky, čímž se cyklus uzavře. Během přidávání nových či úpravy kol existujících je možné režim opustit bez ukládání linky. V tomto režimu úprav se také přidává jízdní řád, to ve formě ukládání času jízdy mezi zastávkami v minutách.

4.2. Formát souboru okolí

Soubor okolí má příponu .etmap („Eurotram map“) a je sestaven z hlavičky, udávající základní informace a seznamů objektů. Hlavička má následující formát:

ETMP V D N I T Y S R P L G B M W F X J...J0 A...A0 E

Všechny soubory tohoto formátu musí začínat 4 byty odpovídajícími textu „ETMP“ v kódování ASCII. Dále následuje 16-bitové číslo *V* označující verzi formátu číslem od 0 do 65535 ($2^{16} - 1$). Program musí povinně ukázat chybu je-li číslo verze odlišné od verze popřípadě verzí kterou může program přečíst. *D* je 64-bitové číslo označující unixový čas v milisekundách při poslední změně souboru. Čísla *N*, *I*, *T*, *Y*, *S*, *R*, *P*, *L*, *G*, *B*, *M*, *W*, *F* a *X* jsou 32-bitová a určují počet jednotlivých typů objektů (*N* určuje počet bodů, *I* počet výhybek, *T* tratí a staničních tratí, *Y* výhybkových návěstidel, *S* křižovatkových návěstidel a předzvěstí, *R* rádiových přijímačů, *P* staničních sloupů, *L* pouličních lamp, *G* stromů, *B* budov, *M* památek, *W* stěn, *F* značek a *X* parcel) v scénáři. *J* a *A* jsou textové řetězce u libovolné délce označující jméno scénáře a autora mapy. Oba řetězce jsou zakončeny nulovým znakem. 8-bitové číslo *E* určuje počet pixelů na metr.

Seznam objektů je za sebou uspořádaný dle typů. V rámci typů nejsou avšak nijak tříděny. Poloha (označené v následujících ukázkách jako čísla *X*, *Y*) a výška (označená v následujících ukázkách jako *H*) jsou vždy uloženy jako 32-bitová čísla. Rotace (označená v následujících ukázkách jako *R*) je vždy uložena ve stupních a je uložena jako 16-bitové číslo. Staniční kódy (označené v následujících ukázkách jako *SSSS*) jsou vždy uloženy jako 4-písmenné řetězce uloženy ve formátu ASCII které je možné jednoduše převést na a interpretovat jako 32-bitová čísla. Staniční kód by měl být pro každou stanicí unikátní. Stanici s několika stanovišti je přiřazen jeden kód. Nemá-li objekt přiřazený staniční kód, je vykreslován. Památky jsou vždy vykreslovány.

Body jsou uloženy v následujícím formátu: *X Y H SSSS BBBB*

Nejen *SSSS*, ale také *BBBB* jsou staniční kódy označující stanici ke kterému bod náleží a stanici se kterou bod hraničí. Pokud bod nehraničí se žádnou jinou stanicí, jsou všechny byty *B* nulové.

Výhybky jsou uloženy v následujícím formátu: *X Y H SSSS B L F P X G T*

32-bitové číslo *B* určuje pořadí bodu který slouží jako trať směřující k výhybce. Směry odboček jsou považovány ze směru bodu v pořadí *B*. 32-bitové číslo *L* označuje bod do kterého vede levá odbočka výhybky. 32-bitové číslo *F* označuje bod do kterého vede výhybka, je-li nastavena rovně. 32-bitové číslo *P* označuje bod do kterého odbočuje prvá odbočka výhybky. Čísla *B*, *L*, *F* a *P* se nesmí rovnat pořadí výhybky. Pokud je číslo *L*, *F* a *P* rovno -1, výhybka do daného směru neodbočuje. Jsou-li *L*, *F* i *P* rovny -1, je výhybka neplatná a při jejím průjezdu ukončen současný scénář. Je-li číslo *B* rovné -1, je výhybka také neplatná. 32-bitové číslo *X* určuje pořadí rádiového přijímače výhybky. 32-bitové číslo *G* určuje pořadí návěstidla výhybky. Písmeno *T*, uložené v kódování UTF-8, označuje písmeno v návěstidle označující tuto výhybku.

Tratě jsou uloženy následovně: TK M N F A B C D H1 . . . 9

První 2 byty odpovídají textu „TK“ v kódování ASCII. 32-bitová čísla M a N určují pořadí prvního a druhého bodu které trať spojuje. 8-bitové číslo F určuje flagy. V objektu jsou uloženy 3 flagy – zda je první bod výhybka či ne, zda je druhý bod výhybka či ne a zda je trať přímka či Beziérova křivka. Flagy jsou uloženy v uvedeném pořadí. 32-bitová čísla A , B určují polohu prvního kontrolního bodu Beziérovky křivky, zatímco 32-bitová čísla C , D určují polohu druhého bodu. Pokud je trať přímka, jsou tyto hodnoty nastavené na nulu. Čísla $H1$ až $H9$ určují deset výškových bodů tratě, rovnoměrně rozprostřených podél délky tratě. Jelikož změna výšky může být také desetinným, jsou výškové body stonásobkem jejich opravdových hodnot. Tratě nesmí vést mezi 2 výhybkami. Každý bod který není výhybkou smí mít maximálně 2 tratě. Všechna čísla jsou 32-bitová.

Staniční tratě jsou uloženy následovně: ST M N F A B C D H1 . . . 9 SSSS

První 2 byty odpovídají textu „ST“ v kódování ASCII. Staniční tratě ukládají stejnou informaci jako tratě a svůj staniční kód, který určuje o jakou stanici se jedná. Tratě a staniční tratě se rozlišují na základě prvních 2 bytů. Tratě a staniční tratě jsou uloženy společně.

Výhybková návěstidla jsou uloženy následovně: X Y H R SSSS N L . . . L

8-bitové číslo N určuje počet jednotek návěstidel. Každá jednotka je identifikována svým písmenem L . Každé písmeno je uloženo jako 8-bitové číslo v kódování UTF-8.

Křižovatková návěstidla jsou uloženy následovně: JS X Y H R P SSSS

První 2 byty odpovídají textu „JS“ v kódování ASCII. P je 16-bitové číslo označující výšku sloupu v metrech. Je-li P rovno nule, je křižovatkové návěstidlo vykreslováno pod trakčním vedením, na stejné úrovni jako značky. Výška ve kterém je vykreslováno návěstidlo se rovná součtu H a P .

Předvěsti pro křižovatková návěstidla jsou uloženy následovně: PJ X Y H R P I SSSS

První 2 byty odpovídají textu „PJ“ v kódování ASCII. Stejně jako u křižovatkového návěstidla, P je 16-bitové číslo označující výšku sloupu v metrech. 32-bitové číslo I označuje pořadí návěstidla ze kterého předvěst čerpá informaci. Předvěsti a křižovatková návěstidla jsou uloženy společně.

Rádiové přijímače a stromy jsou uloženy následovně: X Y H SSSS

Staniční kód radiového přijímače musí odpovídat staničnímu kódu jeho výhybky.

Staniční sloupy a pouliční lampy jsou uloženy následovně: X Y H R SSSS

Budovy jsou uloženy následovně: X Y H R T SSSS

Číslo typu	Název typu	Popis	Šířka, hloubka
0	Normální	Standardní městská budova, typická pro centrum.	20m, 19m
1	Rohová	Rohová městská budova, typická pro centrum.	24m, 24m
2	Rodinná	Rodinná vila, typická pro Hanspaulku.	20m, 12m
3	Panelová	Vysoká panelová budova, typická pro sídliště.	25m, 15m

Tabulka 6: Seznam typů budov, jejich popisků a rozměrů.

Číslo T je 8-bitové a určuje typ budovy dle následující tabulky.

Památky jsou uloženy následovně: X Y H R T

Číslo T je u památek 16-bitové a určuje typ budovy dle následující tabulky. V simulátoru jsou obsaženy pouze ty památky, které jsou podél tratě vidět, nikoliv všechny které je možné v Praze nalézt.

Kód památky	Název památky	Typ budovy	Šířka, hloubka
OLHB	Olšanské hřbitovy	Administrativní	TODO
ATRF	Atrium Flora	Obchodní centrum	TODO
JZPC	Kostel na nám. Jiřího z Poděbrad	Kostel	TODO
NTM1	Národní muzeum, stará budova	Muzeum	TODO
NTM2	Národní muzeum, nová budova	Muzeum	TODO
NMMC	Kostel na nám. Míru (sv. Ludmila)	Kostel	TODO
VRSC	Zámek na Vršovickém nám.	Historická	TODO
CCHC	Kostel na Čechovo nám.	Kostel	TODO
ZZTV	Žižkovská televizní věž	Historická	TODO

Tabulka 7: Seznam památek, jejich popisků a rozměrů.

Památky neukládají sounáležitost ke stanici jelikož jsou vždy vykreslovány.

Stěny jsou uloženy následovně: A B I C D J H SSSS M . . . M0

Čísla A , B a C , D určují polohu obou bodů stěny. 16-bitové číslo H určuje výšku stěny. M je textový řetězec určující název materiálu stěny.

Značky, odborně návěsti, jsou v souboru uloženy následovně: X Y H R SSSS T

Kde T je 16-bitové číslo určující typ návěsti. Seznam všech návěstí, jejich funkcí, čísel i vzhledu je možné najít v kapitole „Pohyb po trati a návěsti“.

Parcely jsou uloženy následovně: A B C D E F G H I J K L SSSS TTTT
M . . . M0

32-bitová čísla A až L jsou hodnoty jednotlivých bodů. Hodnoty jsou uskupeny podle náležitosti k bodu (tj. $X_1, Y_1, H_1 \dots X_4, Y_4, H_4$) a určují polohu a výšku všech čtyř vrcholů parcely. Jsou také uloženy 2 staniční kódy (S, T) a název materiálu parcely M . Dle názvu materiálu najde simulátor ve svém seznamu odpovídající materiál a vykreslí parcelu s ním. Body jsou uloženy po směru hodinových ručiček.

Všechna čísla jsou se znaménkem a podporují záporné hodnoty za pomoci dvojkového doplňku. Všechny hodnoty jsou uloženy v pixelech a jsou převedeny na metry za pomoci hodnoty uložené v hlavičce souboru. Mezery jsou čistě grafickým dodatkem a nejsou ve formátu (mimo řetězce) nikdy obsaženy. Čísla jsou uloženy ve formátu little-endian.

4.3. Formát souboru scénáře

Soubor scénáře má příponu .etscr („Eurotram scenario“) a je sestaven z hlavičky a jízdního řádu.

Hlavička souboru má následující formát:

ETSC V D S W N J . . . J0 A . . . A0

Všechny soubory tohoto formátu musí začínat 4 byty odpovídajícími textu „ETSC“ v kódování ASCII. 64-bitové číslo *D* určuje unixový čas. 64-bitové číslo *S* určuje počáteční čas scénáře v unixovém čase. *S* udává čas herní, nikoliv reálný. Simulátor nemusí hodnotu *S* dodržovat, může si nastavit vlastní či změnit zadanou. *W* je 16-bitové pole ukládající flagy určující počasí během scénáře. *W* také obsahuje informaci o ročním období ve kterém se scénář odehrává. Současně není změna počasí během scénáře podporována. Roční období a časy východu či západu slunce jsou vypočítány na základě informací obsažených v *S*. Textový řetězec *J* označuje jméno scénáře. Textový řetězec *A* označuje jméno autora. Oba řetězce jsou zakončeny null terminátorem. Při odstranění jakéhokoliv bodu na mapě v nástroji pro úpravu map a scénářů jsou všechny scénáře neodpovídající mapě a musí být vytvořeny znova, a to z důvodu změn pořadí bodů. Přidávání nových bodů či jakékoliv operace s objekty kromě bodů, výhybek a tratí nemají na soubory scénáře vliv.

N je 8-bitové číslo označující počet kol od konečné do konečné které jsou v scénáři obsažené. Každé kolo má hlavičku, seznam výhybek a seznam zastávek. V hlavičce je obsaženo číslo linky, čas odjezdu z konečné a typ jízdy – jedná-li se o jízdu služební, výjezd z/zájezd do vozovny či normální kolo na lince. V seznamu zastávek a výhybek jsou obsaženy čísla a směry výhybek, kódy zastávek a jízdní řád zastávek jejichž územím tramvaj projíždí.

Hlavička kola má následující formát: L M I N F T

L je 8-bitové číslo označující číslo linky. *M* je 8-bitové písmeno v kódování UTF-8 určující druhé písmeno linky. Je-li linka číselná, je tato hodnota nastavena na 0 a je použita pouze hodnota v *L*. Je-li linka označena písmeny (např. XA, XC, X5 apod.) je tato hodnota nenulová a *L* je považováno jako první písmeno v lince kódované ve formátu UTF-8. Je-li *M* nastaveno na jinou hodnotu, je linka považována manipulační (*M* = 1), služební (*M* = 2), zvláštní (*M* = 3), cvičnou (*M* = 4) či zkušební (*M* = 5) a hodnota *L* je ignorována. V závislosti od typů linky jsou jinak vykreslovány panely. *I* je 16-bitové číslo označující pořadové číslo spoje (formát čísla je detailněji popsáno v kapitole TODO). 16-bitové číslo *N* označuje počet stanic pro dané kolo. 32-bitové číslo *F* označuje pořadí prvního bodu kterým vozidlo projíždí. *T* je písmeno v kódování UTF-8 které je nastaveno na ‚N‘ je-li první bod standardní bod nebo na ‚S‘ je-li první bod výhybkou.

Seznam zastávek a výhybek je zamíchaný. Uloženy jsou nejen standardní zastávky, ale i také služební zastávky na obratištích či ve vozovnách. Pro každou zastávku je uložen počet výhybek které tramvaj musí zajistit správný směr (nikoliv pouze projet) *N* cestě do

další zastávky. N je uloženo jako 8-bitové číslo. První položka je vždy zastávka. Zastávky jsou tím pádem uloženy následovně: S I CCCC T M K N

První byte odpovídá písmenu „S“ v kódování ASCII. I je 32-bitové číslo určující pořadí staniční tratě na mapě. C je staniční kód zastávky v kódování ASCII. T je 16-bitové číslo určující čas od minulé zastávky, a to v minutách. M je 16-bitové číslo označující délku postoje tramvaje na zastávce, a to v minutách. Je-li postoj tramvaje na zastávce delší než 3 minuty, může uživatel simulátoru ten čas přeskočit nejpozději do minuty před odjezdem. K je 8-bitové číslo, které je rovno jedné pouze pokud je zastávka kontrolním bodem. Není-li zastávka kontrolním bodem, je K rovno nule.

Výhybky jsou uloženy následovně: W I D

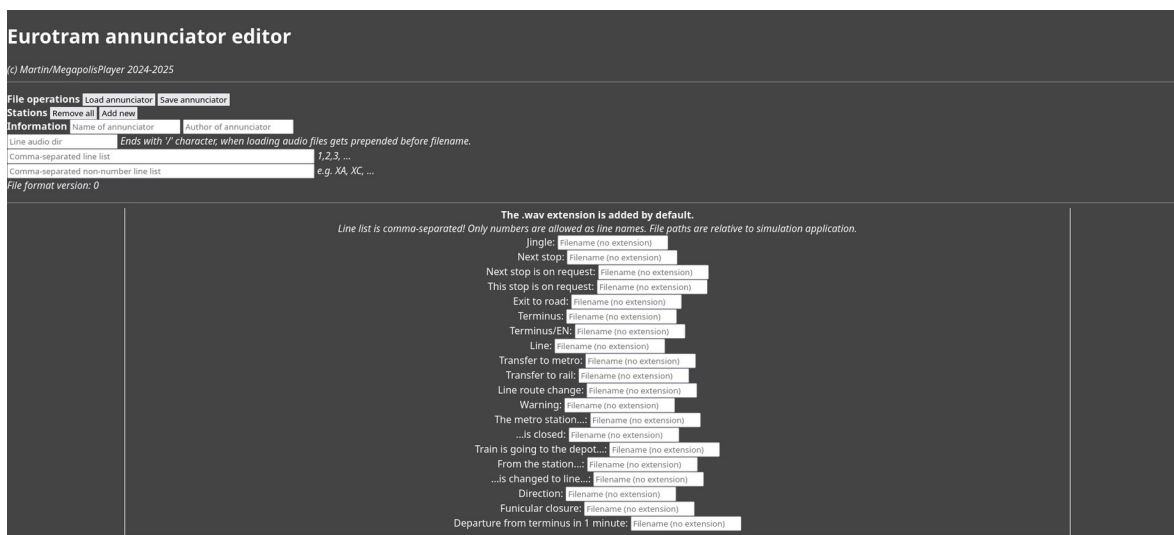
První byte odpovídá písmenu „W“ v kódování ASCII. I je 32-bitové číslo pořadí výhybky na mapě, zatímco 8-bitové číslo D určuje směr přehození výhybky. Je-li $D = 1$, je výhybka přehozena vlevo, je-li $D = 0$ rovně nebo je-li $D = 2$ vpravo. Je-li $D = 3$, najíždí se v dané situaci proti hrotu výhybky a není potřeba jí přehazovat jelikož se přehodí sama. Je-li D rovno jiné hodnotě, výhybka nemá žádný směr a při jejím průjezdu je scénář ukončen. Začíná-li linka na úseku trati hned za výhybkou, není potřeba výhybku přehazovat.

Normální body a tratě nejsou uloženy. Výhybky v simulátoru, stejně jako výhybky na opravdových tratích, zjišťují směr tramvaje pouze když tramvaj již projíždí okolo rádiového přijímače výhybky. Je-li mezi stanicemi několik výhybek, začne tramvaj vysílat jiný signál po přejezdu výhybky za pomoci dodatečného čidla.^[1]

Všechna čísla jsou se znaménkem a podporují záporné hodnoty za pomoci dvojkového doplňku. Mezery jsou čistě grafickým dodatkem a nejsou ve formátu (mimo řetězce) nikdy obsaženy. Čísla jsou uloženy ve formátu little-endian.

4.4. Nástroj na vytváření hlásiče

Nástroj na vytváření hlásiče je, stejně jako ostatní nástroje, webovou aplikací napsaná v programovacím jazyce JavaScript (za pomoci technologií HTML a CSS). Nástroj exportuje soubor hlásiče, který je pak simulátorem načten a použit. Nástroj neslouží k nahrání zvukové informace, nýbrž k odkazování na složku která zvukovou informaci



Obrázek 14: Fotografie nástroje pro vytváření hlásiče při spuštění

ukládá.

V nástroji je možné nastavit stav jednotlivých zastávek – zda jsou přítomny, uzavřeny či omezeny přestupní možnosti na metro nebo na železnici, jaké linky zastávkou projíždí a jaký zvukový soubor obsahuje její název. Soubory hlásiče jsou rozděleny na dvě části – základové, neboli ty které nesouvisí ze zastávkami a zastávkové, které jsou pro každou zastávku různé. V kategorii zastávkových jsou uloženy jednotlivé názvy zastávek. V kategorii nesouvisejících jsou uloženy obecná hlášení, například hlášení pro příští zastávku nebo pro konečnou zastávku. Nástroj umí soubory hlásiče jak vytvářet a ukládat, tak i načítat a upravovat. V nástroji je také možné nastavit jaké linky mají hlášení přítomné v simulátoru.

Zvuky jednotlivých vozů jsou uloženy v jejich konfiguračních souborech. Nástroj slouží výhradně ke tvorbě hlášení informací pro cestující, která jsou ve všech typech vozů stejná.

4.5. Formát souboru hlásiče

Soubor hlásiče má příponu .etanc („Eurotram announcements“) a je sestaven z hlavičky, seznamu základních hlášení a seznamu zastávek. Zastávky nejsou nijak tříděny.

Hlavička je uložena následovně:

```
ETAN V D J...J0 A...A0 P...P0 K L...L M (N...N0)...(Ni...Ni0)
Z
```

První 4 byty odpovídají textu „ETAN“ v kódování ASCII. *V* je 16-bitové číslo označující verzi formátu souboru. *D* je unixový čas reprezentovaný jako 64-bitové číslo. Řetězec *J* označuje název hlásiče. Řetězec *A* označuje autora hlásiče. Řetězec *P* označuje složku ve které jsou obsaženy zvukové soubory označující linky definované v *L* a *N*. Všechny tři řetězce jsou uloženy v kódování UTF-8. *K* označuje počet linek *L*, pro které jsou v simulátoru obsaženy hlášení. *K* i *L* jsou 8-bitová čísla. 8-bitové číslo *M* označuje počet řetězců *N* označujících názvy nečíselných linek. *Z* je 32-bitové číslo označující počet zastávek.

Po hlavičce je v souboru obsažena informace o základních hlášeních. Každé takovéto hlášení je uloženo jako řetězec znázorněný jako *F* sloužící jako odkaz na soubor hlášení. Pořadí hlášení je jasné dané a je následující: znělka, hlášení příští zastávky, hlášení informující že příští zastávka je na znamení, hlášení informující že tato zastávka je na znamení, hlášení výstupu do vozovky, hlášení konečné zastávky v českém a anglickém jazyce, hlášení přestupu na metro, hlášení přestupu na vlaky (linky *S*), hlášení o změně trasy linky, hlášení upozornění pro cestující, hlášení „Stanice metra...“ a „...je uzavřena.“, hlášení jízdy do vozovny („Vlak jede do vozovny“), hlášení a „ze zastávky“ a „je dále vedena jako linka“, slovo „směr“, hlášení o uzávěru lanové dráhy na Petřín a služební hlášení odjezdu z konečné za jednu minutu.

Každá zastávka je uložena následovně:

```
SSSS N...N0 A...A0 K L...L M (N...N0)...(Ni...Ni0) F
```

S je 4-bytový staniční kód v kódování ASCII, řetězec *N* je název stanice. Řetězec *A* je název souboru zvukového hlášení. Složka je udána jako relativní k souboru simulátoru. Dále následuje *K* 8-bitových čísel znázorňující linky projíždějící zastávkou *L*. *K* je číslo 8-bitové. 8-bitové číslo *M* označuje počet řetězců *N* označujících názvy nečíselných linek, které zastávkou projíždí. *F* je 8-bitové bitové pole označující flagy zastávky – zda je přítomen přestup na metro, přestup na vlaky (hlášení „přestup na linky *S*“) a zda je potřebné hlásit uzávěr lanové dráhy na Petřín. Všechny řetězce jsou zakódované ve formátu UTF-8.

Ročníková práce

Fyzicky realistický simulátor pražských tramvají včetně okolí bodovacího systému

Všechna čísla jsou se znaménkem a podporují záporné hodnoty za pomocí dvojkového doplňku. Mezery jsou čistě grafickým dodatkem a nejsou ve formátu (mimo řetězce) nikdy obsaženy. Čísla jsou uloženy ve formátu little-endian. Cesty k souborům jsou relativní a jsou procházeny od umístění aplikace simulátoru. Koncovky souborů nejsou ukládány – všechny zvukové soubory jsou ve formátu WAV.

5. Implementace simulací

5.1. Infrastruktura působení sil na vozidla

Fyzikální simulace jsou aplikovány pro každé současně viditelné vozidlo. Jelikož mají normální a „grafické“ snímky mezi sebou nepravidelné intervaly, je přidán systém tzv. „fyzikálního snímku“, který volá funkci zpracovávající fyzikální simulace každých 20 milisekund neboli 50-krát za sekundu. Tato hodnota je v kódu simulátoru nastavitelná. Pro každé vozidlo se každý fyzikální snímek spustí funkce vypočítávající všechny simulované veličiny. Vozidla ukládají svojí hmotnost, současnou rychlost, plochu jednotlivých stran, maximální výkon motorů, celkovou spotřebovanou energii ale i jiné fyzikální hodnoty potřebné pro výpočty.

Na vozidlo jsou aplikovány 2 výsledné síly – pohybová (působící ve směru jízdy, síla pohybuje brzdí tramvaj) a vertikální (použitou pro simulaci vykolejení). Ostatní síly jsou uloženy pro účely výpočtů těchto výsledných.

Pohybová síla se skládá ze síly motorů a společné odporové síly. V simulátoru je také vypočítávána maximální odporová síla F_{rmax} , která je rovna podílu součinu rychlosti v a hmotnosti m , dělenou intervalem volání fyzikální funkce $t = 1/50$ s.

$$\begin{aligned}v &= at \\v &= \frac{F_{rmax}}{m} t \\F_{rmax} &= vm/t\end{aligned}$$

Síla F_{rmax} určuje potřebnou sílu pro zabrzdění tramvaje za jedno volání funkce, neboli 1 padesátinu sekundy. To znamená, že odporové síly jednoduše nemohou překračovat tuto sílu, jelikož by začali zrychlovat tramvaj do opačného směru.

Síly a fyzikální vlastnosti pro několika-článekové tramvaje či několika-vozové soupravy jsou vypočítávány pouze jednou.

5.2. Tíha

Tíhová síla F_g je součin hmotnosti tramvaje m_t tíhového zrychlení g .

$$\vec{F}_g = m_t \times \vec{g}$$

Tíhové zrychlení je bráno jako $9,81\text{ ms}^{-1}$. Má-li souprava více vozů, je hmotnost vynásobena počtem vozů. Hmotnost 1 vozu tramvaje je možné nastavit v konfiguračním souboru typu vozidla.

5.3. Tření, valivý odpor a prokluz kol

Síla valivého odporu F_r je rovna součinu síly normálové F_n koeficientu valivého odporu c_{rr} .

$$\vec{F}_r = \vec{F}_n c_{rr}$$

Koeficient valivého odporu je v simulátoru vždy nastaven na hodnotu 0,001, což je rovno odporu železných kol na železné koleji.^[29]

Třecí síla F_t je rovna součinu síly normálové F_n (která je rovna $F_g \cos \theta$, kde θ je rovna úhlu nakloněné roviny které se současně vůz nachází) součinitele smykového tření c_f .

$$\vec{F}_t = \vec{F}_n c_f$$

Změny počasí a jiné změny podmínek mění pouze koeficient tření. Koeficient tření za ideálních podmínek je nastaven na 0,5. Koeficient tření za ideálních podmínek během klidu vozidla je nastaven na 0,7.^[30*]

Stav	Změna koeficientu tření
Děšť nebo Sníh	-0,2
Náledí (zimní roční období)	-0,3
Listí (podzimní roční období)	-0,2
Pískování	+0,3

Tabulka 8: Efekt počasí různých stavů na koeficient tření

Třecí síla je aplikována pouze při nouzovém brzdění vozidla kolejnicovou brzdou. Při nastavení řadiče jízdy do záporné pozice používá vozidlo brzdění regenerační, které je popsáno detailněji kapitole 5.7 „Elektrické systémy vozidel a celková práce vozidla“.

Prokluz nastává kdy výsledná pohybová síla je ostře větší než síla třecí.^[31] V takovém případě simulátor neaplikuje dále žádnou sílu dokud není prokluz vyřešen.

5.4. Síla působení motorů a rychlost vozidla

Při jízdě vozidla je vypočítávána jeho síla působení motorů. Výslednice této síly, síly třecí a síly tíhové tvoří celkem výslednou pohybovou sílu.

Síla působení motoru F_f , jinak označována jako trakční úsilí, je rovna podílu součinu příkonu motorů P a jeho účinnosti e a současné okamžité rychlosti v .

$$\vec{F}_f = Pe/v$$

Je-li okamžitá rychlost rovna nule, není možné důvodů dělení nulou tento vztah použít. Pro sílu motoru při rozjezdu, který je zde definován jako $v < 3 \text{ m/s}$, jsou použity následující vztahy:

$$M = P/\omega$$

$$\vec{F}_f = \frac{M \eta p^{[32]}}{r}$$

M je točivý moment motoru, který je roven podílu výkonu motoru a jeho úhlové rychlosti ω . Výsledná síla působení motoru je potom vypočítána vynásobením točivého momentu motoru, jeho účinnosti (v simulátoru vždy rovna 95%), hodnoty poměru převodů a následným vydělením tohoto součinu poloměrem kol.

Výsledná rychlost pohybu vozidla je určena na základě 2. Newtonova pohybového zákona, dle kterého sílu je možné zapsat jako součin hmotnosti zrychlení, což lze vyjádřit také rovnicí $F = ma$. Zrychlení se následně převede na okamžitou rychlost v , která je následně poslána do grafické části simulátoru.

5.5. Síly v oblouku a vykolejení

Během zatáčení vozu oblouku je důležitý poměr mezi silou vertikální V a silou boční L .

$$n_f = \frac{\vec{L}}{\vec{V}}$$

Je potřeba také vypočítat tzv. „hraniční poměr“ po kterém se kolo pohybuje buď nahoru po kolejnici nebo po ní sjíždí zpátky. Tento poměr bývá označován jako Nadalův. Rovnice určující stav při nejvyšším L při kterém se kola vertikálně nepohybují se jmenuje Nadalova rovnice je definována následovně: ^[33]

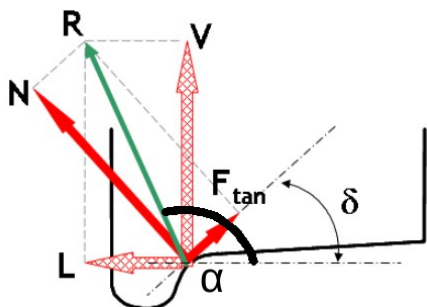
$$\frac{\vec{L}}{\vec{V}} = \frac{\tan(\delta) - c_f}{1 + c_f \times \tan(\delta)} \quad [33]$$

L je vertikální síla, která je rovna vertikální složce síly tíhové, V je síla boční, v simulátoru dostředivá.

Výsledná síla nadnesení kola nad kolejnicí F_{tan} je rovna součinu výsledné síly, kosinu rozdílu úhlu doteku a úhlu vektoru síly k vodorovné ose α , která je definována jako součet sil a V . Vertikální složka této síly je vypočítána jako součin síly F_{tan} kosinu úhlu $90 - \delta$.

$$\vec{R} = \vec{L} + \vec{V} \quad [33]$$

$$F_{tan} = \vec{R} \cos(|\alpha - \delta|) \cos(90 - \delta)$$



Obrázek 15: Hodnoty použité při výpočtu síly nadzvednutí vozidla v oblouku ^[32]

Pokud výška kol nad trati překročí výšku trati, nastavenou v simulátoru jako 20 centimetrů, považuje ze vozidlo za vykolejené. Při vykolejení se program ihned vypne a vypíše hlášku o vykolejení vozidla.

5.6. Aerodynamická odporová síla

Aerodynamická odporová síla je vypočítána z následujícího vztahu:

$$\vec{F}_{odporová} = \frac{1}{2} \rho v^2 c S^{[34]}$$

v je okamžitá rychlost vozidla. V obou rovnicích určuje ρ hustotu vzduchu (která je rovna 1kg/m^3), c činitele odporu (definovaným konfiguračním souborem vozidla) a S plochu průřezu čela vozidla v metrech čtverečných. Hodnota S je vypočítána z výšky a šířky uvedené v konfiguračním souboru. Síla bočního větru je zanedbána.

5.7. Elektrické systémy vozidel celková práce vozidla

U elektrických systémů je simulován kontakt pantografu s trolejovým vedením, současné napětí v úseku sítě trolejového vedení, spotřeba elektrické energie tramvají a celková spotřeba elektrické energie za celou jízdu.

Pro výpočet spotřeby energie je použit následující vztah:

$$E = P/t$$

P je rovno výkonu motoru v danou chvíli. Je vypočítáno jako maximální výkon motorů vynásobený procentuální zátěží nastavenou řidičem $c \in \langle -1; 1 \rangle$. Čas t je konstantní a je roven časovému intervalu mezi jednotlivými voláními funkcí fyzického výpočtu, který je roven právě $1/50$ sekundy. Pro každý fyzikální interval se energie přičítá. Při brzdění, které je simulátoru zjištěno dle záporného příkonu, se energie vrací zpátky, a to s účinností 75% ($\eta = 0,75$).

Pro výpočet současného proudu je využit Wattův zákon.

$$P = VI \cos \phi$$

V je současné napětí, I je současný proud a ϕ je fázový úhel střídavého proudu, který je simulátoru nastaven na $0,9$ radiánů. Současné napětí je nastaveno na napětí sítě, neboli v Praze 600V .^[5]

Celková energie využita vozidlem je později použita pro výpočet celkových emisí, viz kapitola 5.10 „Výpočet a srovnávání emisí“.

5.8. Osvětlení

Pro simulaci osvětlení je použita modifikovaná verze Blinn-Phongova modelu odražení světla. Osvětlení Blinn-Phongova modelu je možné rozdělit na části – světlo okolní (angl. ambient), světlo difuzní (angl. diffuse) a světlo odražené (angl. Specular). Blinn-Phongův model je pouze malá úprava původního Phongova modelu. Jediná změna se týká výpočtu odraženého světla, a tím je přidání polovičního vektoru pro lepší zobrazení při velkém rozdílu uhlů vektoru vzhledu a vektoru světla.^[20 p. 115-125 *]

V standardní verzi modelu jsou přítomny pro každou část vlastní světelné hodnoty. Z důvodu způsobu exportování souboru modelovacím programem, je pro každé těleso nastavena pouze barva základní.

Výsledná barva fragmentu o je rovna součinu základní barvy $c_{r,g,b}$ a větší hodnoty (funkce $\max(a, b)$ – vrací „maximum“ z dvou hodnot) mezi hodnotami světla okolního a a součtu sum výsledků všech n světél – sum jejich difúzních ($d_{r,g,b}$) a odrazových částí $s_{r,g,b}$.

$$[o_r, o_g, o_b] = [c_r, c_g, c_b] \times \max\left(a, \sum_{i=1}^n [d_r, d_g, d_b] + \sum_{i=1}^n [s_r, s_g, s_b]\right)$$

$$\text{kde } a \in \langle 0, 1 \rangle, c_{r,g,b} \in \langle 0, 1 \rangle, d_{r,g,b} \in \langle 0, 1 \rangle, s_{r,g,b} \in \langle 0, 1 \rangle$$

Barva difuzního světla je součinem sílou difúzního světla p_d , barvou světla $l_{r,g,b}$ a atenuací t_d .

$$[d_r, d_g, d_b] = p_d \times [l_r, l_g, l_b] \times t_d \text{ kde } p_d \in \langle 0, 1 \rangle, l_{r,g,b} \in \langle 0, 1 \rangle, t_d \in \langle 0, 1 \rangle$$

Barva světla odraženého je součinem čtyř částí – sílou odraženého světla p_s , barvou odraženého světla $l_{r,g,b}$, odrazivost materiálu r_m a atenuace světla t_s .

$$[s_r, s_g, s_b] = r_m \times p_s \times [l_r, l_g, l_b] \times t_s \text{ kde } r_m \in \langle 0, 1 \rangle, l_{r,g,b} \in \langle 0, 1 \rangle, t_s \in \langle 0, 1 \rangle$$

Síla difúzního světla p_d je skalárním součinem normalizovaných normálů n_f obličeje na kterým fragment leží a normalizovaného vektoru u směřujícího na světlo. Součin je zdola omezen na hodnotu 0 funkcí \max . Výpočet je graficky znázorněn na následujících obrázcích.

$$p_d = \max(\hat{n}_f \cdot \hat{u}, 0)$$

Síla odraženého světla p_s je skalárním součinem normalizovaných normálů n_f obličeje na kterým fragment leží a středového vektoru h . Vektor h je roven normalizovanému součtu vektoru směřujícího na světlo u a vektoru směřujícího od pohledu uživatele v . Výsledek je také umocněn na exponent s_m uvažující hodnotu lesku materiálu. Výpočet je graficky znázorněn na následujících obrázcích.

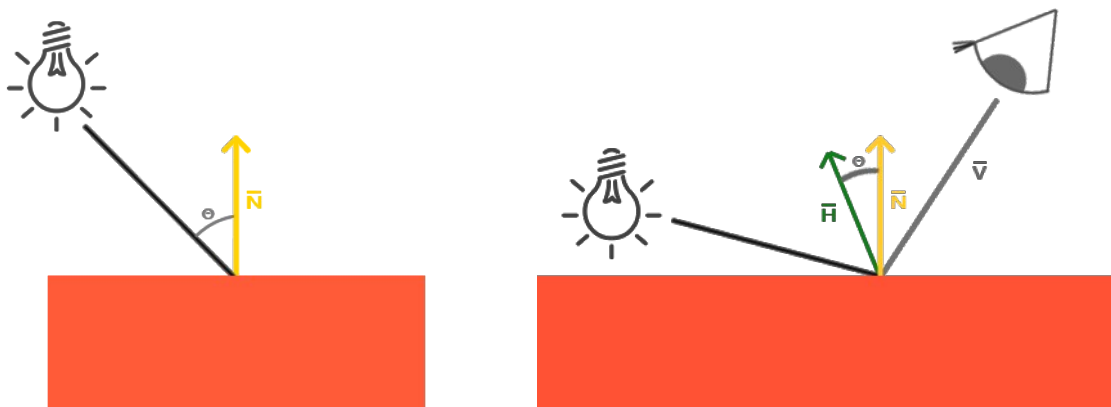
$$\vec{h} = \vec{u} + \vec{v}$$

$$p_s = \max(\vec{h} \cdot \vec{n}_f, 0)^{s_m}$$

Odrazivost materiálu r_m je vypočítávána z tzv. indexu lomu („Index of Refraction“, IOR) označeného jako i_r a to následující rovnicí:

$$r_m = ((i_r - 1) / (i_r + 1))^2 * 12,5 \text{ kde } i_r \in \langle 1, \infty \rangle^{[35]}$$

Odrazivost je následně omezena na interval $r_m \in \langle 0, 1 \rangle$ pro splnění podmínky rovnice pro výpočet barvy světla odraženého.



Obrázek 16: Znázornění výpočtu difúzního světla dle (Blinn-)Phongova modelu.

Obrázek 17: Znázornění výpočtu odraženého světla dle Blinn-Phongova modelu.

Obrázky jsou uspořádány dle čísla. Obrázek z menším číslem je vlevo, z větším vpravo.

Zdroj prvního obrázku: 20 p. 116, druhého: 20 p. 122

Atenuace světla je vypočtena následující rovnicí:

$$t_x = \frac{1}{c + ld + qd^2} \text{ kde } t_x \in \langle 0, 1 \rangle, c \in \langle 1, \infty \rangle, l, q \in \langle 0, \infty \rangle, d \in \langle 0, 1 \rangle^{[20 \text{ p. } 140]}$$

Dosah světla (v bezrozměrných jednotkách)	Konstantní parametr c	Lineární parametr l	Kvadratický parametr q
7	1	0,7	1,8
20	1	0,22	0,2
50	1	0,09	0,032
100	1	0,045	0,0075
200	1	0,022	0,0019

Tabulka 9: Hodnoty pro výpočet atenuace závislosti na dosahu světla.^[20 p. 141]

V rovnici jsou c , l a q konstanty určené pro danou vzdálenost světla z předchozí tabulky. d označuje vzdálenost. t_x označuje výslednou atenuaci.

V simulátoru jsou implementovány 3 typy světel – světlo směrové (angl. Directional), světlo bodové (angl. Point) a světlo reflektorové (angl. Spotlight). Síla odraženého světla se u všech typů světel vypočítává stejně.

Světlo směrové je z tří typů nejjednodušší. Směr světla je předem nastaven programem a je vždy stejný. Pro směrové světlo také není vypočítávána atenuace, jelikož směrové světlo nemá definovanou žádnou polohu. ^[20 p. 140 *]

Směrový vektor pro světla bodová a reflektorová je roven normalizovanému rozdílu polohy světla a lokace fragmentu. ^[20 p. 137 *]

Světlo bodové se od směrového liší využitím vypočítaného směrového vektoru a vypočítáváním atenuace světla.

Světlo reflektorové je ze tří typů pro výpočet nejtěžší. Obsahuje veškeré výpočty světla bodového a k tomu vypočítává i sílu světla v závislosti od vnitřního a vnějšího úhlu kuželu. Sílu reflektorového světla p je možné vyjádřit následující rovnicí, kde úhel a určuje úhel mezi vektorem směru světla d a vektorem ukazujícím na polohu světla z pohledu fragmentu s . Hodnoty i a o určují vnitřní a vnější úhly kuželu reflektorového světla. Vektor směru světla d je otočen (vynásoben -1) aby byl nasměrován do směru světla, nikoliv naopak. ^[20 p. 143 *]

$$a = s \cdot -d$$

$$p = \frac{a - o}{i - o} \text{ kde } o > i, i, o, a \in (0, \frac{1}{2}\pi) \text{ rad} \Leftrightarrow (0, 90) \text{ deg}$$

Rovnice vypočítává podíl rozdílu úhlu a a vnějšího úhlu o a rozdílu vnitřního a vnějšího úhlu. Hodnota v čitateli je vydělena rozdílem vnitřního úhlu a vnějšího úhlu. Hodnota v jmenovateli je vždy kladná, jelikož $i < o$ a tím pádem $\cos(i) > \cos(o)$. Hodnota p je po výpočtu omezena na interval $p \in \langle 0, 1 \rangle$, což znamená že jakákoliv záporná či nulová hodnota v jmenovateli.

Pokud je $a \geq o$ (úhel je větší nebo roven než maximální, vnější úhel světla), a s tím je kosinová hodnota $\cos(a) < \cos(o)$, vychází číselník záporně či nulově, a jelikož je hodnota ve jmenovateli vždy kladná, daný fragment není světlem osvětlen. (Výsledek je záporný což se následně omezí na hodnotu 0.)

Funkce skalárního součin vrací úhel v podobě kosinové hodnoty. Z pohledu časové účinnosti je z toho důvodu jednodušší počítat s kosinovými hodnotami úhlů než s jejich opravdovými hodnotami. Veškeré výpočty probíhají v radiánech.

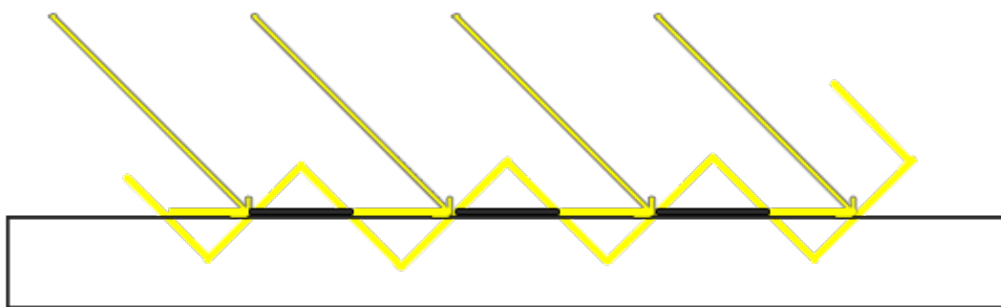
Pro korekci gammy (poměru mezi fyzikálním jasnem a vnímaným člověkem) byl využit hardwarový komponent přes API OpenGL. ^[20 p. 277 *]

5.9. Stíny

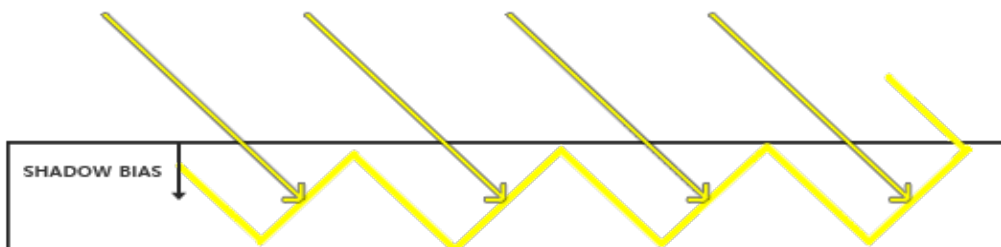
Pro simulaci stínů je použita interní „hloubková mapa“ v API OpenGL. Hloubková mapa určuje které objekty jsou z daného pohledu (například uživatele, světla apod) zakryté a nejsou tím pádem vidět. Hloubková mapa je pro tento účel správným nástrojem jelikož stíny se vždy nachází tam, kam nedosahuje světlo, neboli zakrytá místa z pohledu daného světla. Hloubkové mapy pro každé světlo se vykreslují na hloubkovou texturu, která obsahuje hloubkovou informaci fragmentů jenž dané světlo vidí. Tato hloubková textura ukládá také vzdálenost nejbližšího fragmentu ke světlu. Všechny texturey světel jsou následně poslány do shaderu, kde jsou následně přečteny a aplikovány.

Hloubka současného fragmentu (která je zjištěna aplikováním transformační matice světla na pozici fragmentu uloženou jako čtyřmístný vektor a dělením výsledné pozice posledním parametrem vektoru) je porovnána z hloubkou nejbližšího k světlu fragmentu. Je-li současná hloubka vyšší než hloubka nejbližšího fragmentu, nachází se daný fragment ve stínu. Výsledný stín je průměr stínu fragmentu a jeho okolních fragmentů. Způsoby přidávání plynulejších přechodů stínů se nazývají PCF („percentage-closer filtering“), tento termín avšak zahrnuje velký počet různých způsobů tvorby plynulejších přechodů, a průměrování okolních fragmentů je pouze jeden z nich. [20 p. 283-299 *]

Pro prevenci tzv. „stínového akné“ je aplikována hodnota (tzv. „shadow bias“), posouvající hodnotu hloubky o několik milimetrů dovnitř do stěny. Tímto jsou eliminovány nepřesnosti v modelu, vznikající kvůli ohraničenému rozměru hloubkové textury kvůli čehož několik fragmentů může dostat stejnou přiřazenou hloubku. [20 p. 307 *]



Obrázek 18: Dopad světla pod úhlem bez aplikování posuvné hodnoty



Obrázek 19: Dopad světla pod úhlem po aplikování posuvné hodnoty (tzv. „biasu“)

Pro každé světlo, které má simulovaný stín, je potřeba znovu vykreslit hloubkovou informaci scény z pohledu pozice světla. Toto vykreslování je sice rychlejší než vykreslování normální scény, jelikož se nepočítají výsledné barvy, avšak je to stále náročná operace. Z tohoto důvodu jsou vykreslovány stíny pouze pro nejpodstatnější světla, a to stíny ze světla nebe (světlo pocházející z slunce a měsíce), stíny z ovladatelného osvětlení (baterky řidiče) a předních světel tramvaje.

Stíny jsou simulovány pouze pro směrová a reflektorová světla. Pro světla bodová, která svítí do všech stran rovnoměrně, je potřeba vykreslovat scénu 6 krát a to do každého směru.

5.10. Simulace zvuků

5.10.1. Simulace zvuku motoru a brzd

Simulace zvuků motoru a brzdění tramvají je založena na předem nahraných zvucích rozjezdu a brzdění.^[2] Zvuky jsou přehrávány v závislosti na nastavený výkon motoru. Vždy se přehrává sekce zvuku 200 milisekund dlouhá.

Veškeré zvuky byly vlastnoručně nahrané na noční lince 91.

5.10.2. Simulace zvuku jízdy

Simulace zvuku jízdy je sestavena z několika prvků – zvuku jízdy po kolejnicích na rovných úsecích, zvuky tření při odbočování a ze zvuků přejezdu výhybky.

5.10.3. Simulace jiných zvuků

Simulace ostatních zvuků typu varovného signálu zavírání dveří, zvuků palubního počítače, zvuku znamení řidiči či zvuku zvonění jsou nahrané předem závisí na jednotlivém typu vozu. Jednotlivé souborové cesty zvuků jsou specifikovány v konfiguračním souboru typu vozidla, který je přesněji popsán kapitole 1.3 „Konfigurace vozidel“.

5.10.4 Simulace hlášení

V simulátoru je obsažena simulace hlášení, a to jak hlášení pro cestující, tak i služební. Hlášení, či v některých případech jejich části, jsou nahrány předem a uloženy jako soubory ve formátu WAV. V simulátoru se následovně hrají v předem určeném pořadí vytvořeném dle reálných hlášení. Toto odpovídá reálnému systému hlášení v autobusech i tramvajích^{[1][2]} Hlášení je možné v simulátoru jakkoliv kombinovat či opakovat. Detaily ohledně hlášení na jednotlivých zastávkách jsou uloženy ve speciálním souboru hlásiče jehož formát je popsán v kapitole 4.5 „Formát souboru hlásiče“.

5.11. Výpočet srovnávání emisí

5.11.1. Výpočet emisí automobilové dopravy

Výpočet celkových emisí, které by vyprodukovala individuální automobilová doprava, je roven součinu potřebných automobilů (podílu maximálního počtu cestujících a kapacity automobilu) a délky linky, která je vypočítána sumou všech délek tratí.

$$e_{auto} = \frac{p_{max}}{c_{auto}} s$$

$$s = \sum_0^n s_{trackLength}(i_{track})$$

Emise průměrného automobilu na kilometr činí $400\text{gCO}_2\text{e}$.^[36] Emise automobilové dopravy jsou vypočítány pro různé kapacity – kapacity průměrné (hodnota 1,25, která byla vybrána zcela arbitrárně dle vlastního pozorování – většina automobilů má 1 cestujícího, občas 2) a kapacity maximální, což pro průměrný automobil činí 5 osob.

5.11.2. Výpočet emisí tramvaje

Výpočet emisí tramvaje je vypočten na základě průměrných emisí při produkci elektrické energie a celkové spotřeby elektrické energie tramvají. Celková elektrická spotřeba tramvaje je vypočítávána při fyzikální simulaci. Emise při výrobě 1 kilowatt-hodiny (3 600 000 joulů/J, 3,6 megajoulů/MJ) elektrické energie ČR činí $450\text{gCO}_2\text{e}$. Jsou použita data pro rok 2023.^[37 *]

$$e_{tram} = P / 3600 \text{ J} \times 450 \text{ gCO}_2\text{e} / \text{kWh}$$

5.11.3. Výpočet ušetřených emisí

Ušetřené emise se rovnají rozdílu hodnot emisí automobilové dopravy emisí tramvaje. Je přidán také výpočet ceny energie pro Českou republiku, Polskou republiku a Spolkovou republiku Německo. Cena za kilowatt je aktuální 1. lednu 2024. Ceny jsou vypočítány v eurech a následně převedeny do lokálních měn dle denního kurzu České národní banky (ČNB) aktuálnímu ke 21. březnu 2025.^[39] Cena v polské měně je vypočtena z měny české.

6. Simulace počasí a jeho účinků

6.1. Infrastruktura systému počasí

Infrastrukturu systému počasí je možné podělit na 2 části – část grafická a část fyzikální. Část fyzikální je složena ze změn fyzikálních koeficientů a jejich následné použití při výpočtu sil a rychlostí. Část grafická vykresluje realistický grafický efekt odpovídající stavu počasí. V jakoukoliv dobu může být přítomno několik stavů počasí.

6.2. Denní noční cyklus

Současný čas dne a noci je možné změnit v uživatelském rozhraní simulátoru.

6.3. Počasí

6.3.1. Vítr

Vítr nemá současně žádný vliv na grafickou ani fyzickou podobu simulátoru. Při fyzikální výpočtech je boční vítr zanedbán a vítr nemá značný vliv na aerodynamickou odporovou sílu.

6.3.2. Déšť

Déšť mění koeficient tření v simulátoru.

Déšť také obsahuje grafický efekt kapek, a zešeření oblohy. Počet kapek v simulátoru je volitelný a nastavitelný.

6.3.3. Bouřky

Bouřka se simuluje rychlou změnou barvy oblohy za pomoci přednastavených hodnot. Hodnotu není možné mimo simulátor měnit, je jí možné změnit pouze v kódu.

6.3.4. Mlha

Efekt mlhy je vytvořen snížením viditelného pole za pomoci nastavení parametrů kamery (parametru tzv. „far plane“, neboli vzdálenosti po které se již objekty nevykreslují) a zešeření oblohy. Zešeření oblohy je vytvořeno násobením stavu dne (konstanty okolního světla, viz kapitola 5.7 „Osvětlení“) a výchozí šedé barvy. Existuje také náhodná šance (současně je simulátoru nastavená na 0,1%) že místo mlhy bude aplikována písečná bouře. Při tomto stavu je výchozí barva mlhy nastavena na barvu hnědou.

6.4. Počasí závisující na ročním období

6.4.1. Sníh

Sníh je v simulátoru modifikovanou verzí deště. Využívá stejné grafické infrastruktury jako déšť, a mění stejné koeficienty. Mění také barvu oblohy – nastavuje jí na světlejší.

6.5. Roční období

6.5.1. Jaro a léto

Během jara a léta je simulátor v standardním režimu s jeho standardními texturami. Toto je výchozí stav simulátoru.

6.5.2. Podzim

Během podzimu se mění všechny textury stromů a trávy na odpovídající podzimní textury. Soubory textur použité během podzimního ročního období jsou označeny příponou *Autumn* např. *tp-grassAutumn.png*.

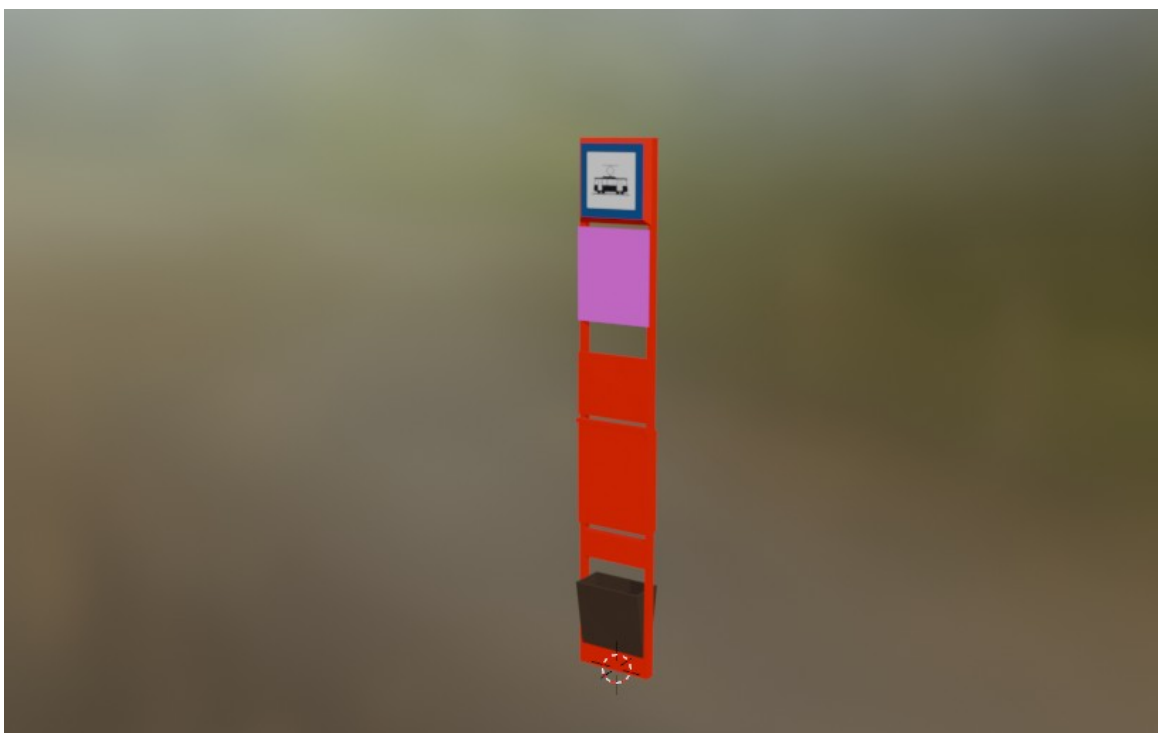
6.5.3. Zima

Během zimy, stejně jako během podzimu, se mění všechny textury stromů a trávy na odpovídající zimní textury. Soubory textur jsou označeny příponou *Winter* např. *tp-grassWinter.png*.

7. Modelování okolí

7.1. Zastávky

Zastávky jsou v simulátoru graficky označeny staničním sloupem. Staniční sloup je modelován dle standardních vnitro-pražských sloupů PID používaných na tramvajových zastávkách.



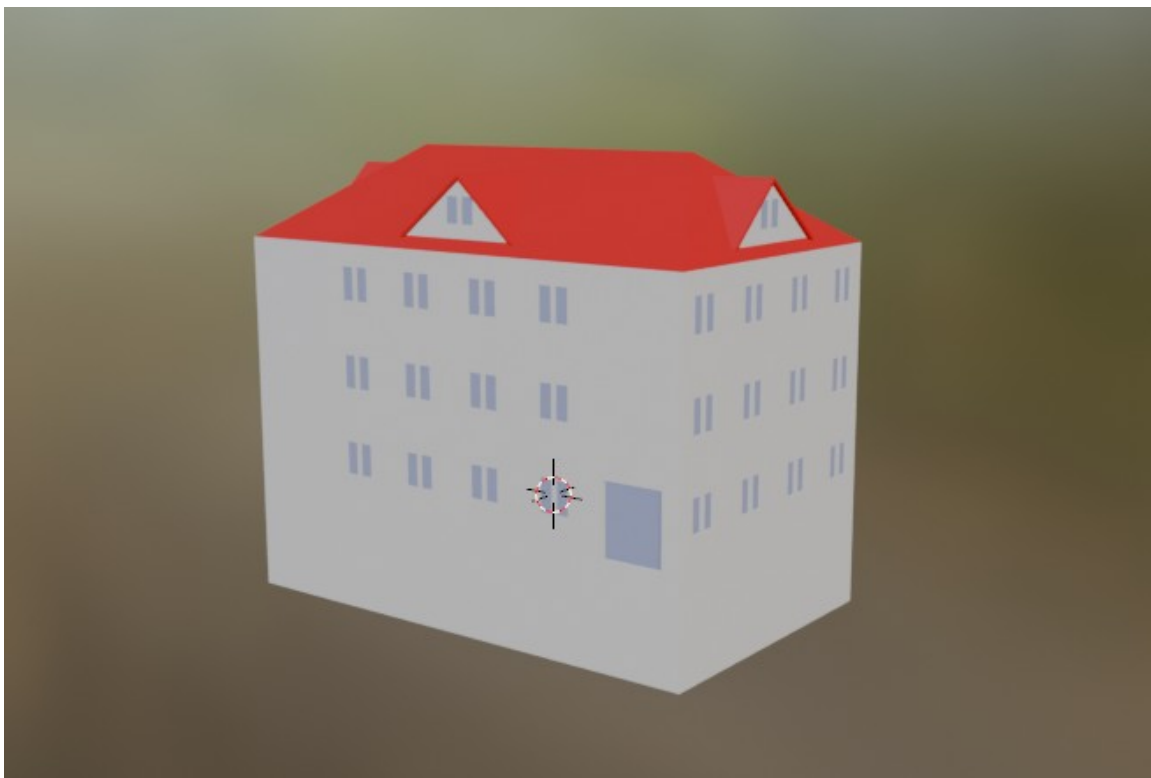
Obrázek 20: Obrázek modelu zastávkového sloupu

7.2. Standardní budovy

7.2.1. Typy budov

Simulátor obsahuje 4 typy obvyklých budov. Každá budova má jiné, avšak předem definované rozměry.

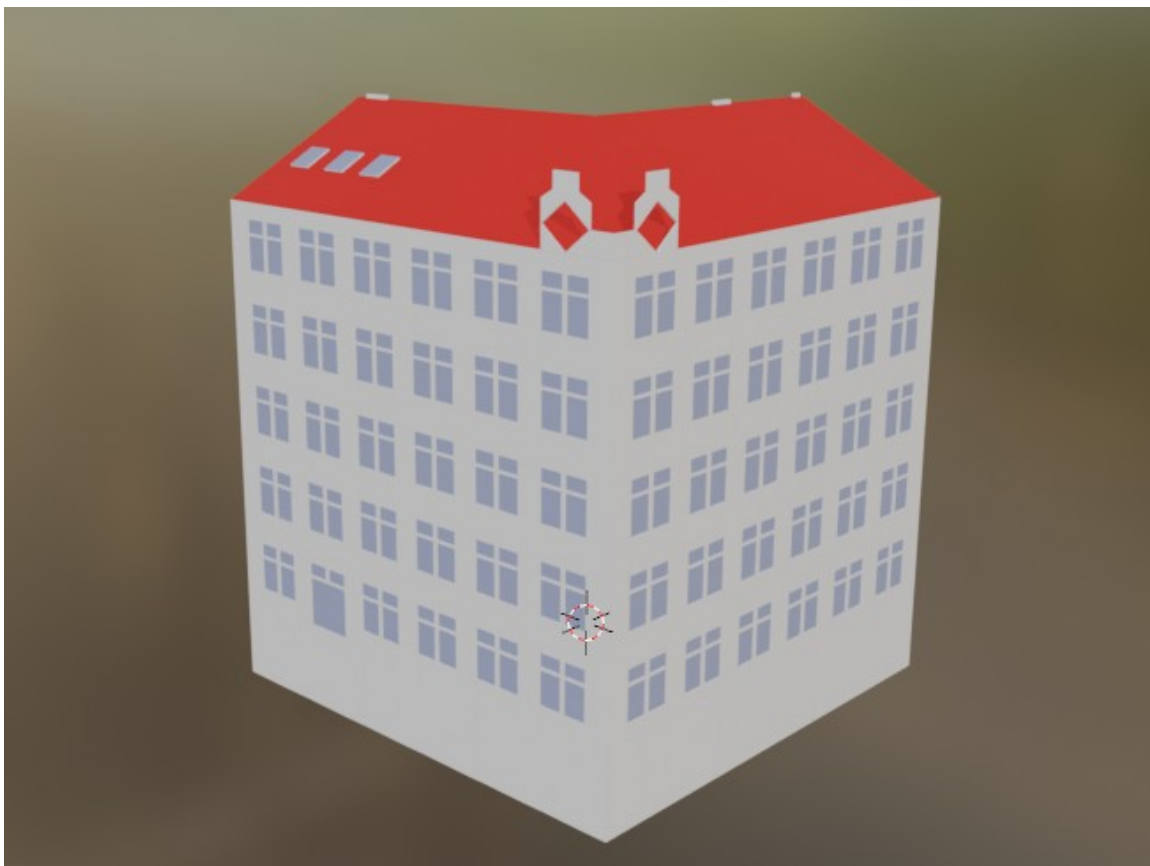
Každá budova, která má okna, musí obsahovat materiál s názvem „Material.windowDay“. Tento materiál je při rozsvícení nočního osvětlení přestaven na „Material.windowNight“. Tato změna povoluje simulovat noční osvětlení uvnitř budov.



Obrázek 21: Obrázek modelu rodinného domku



Obrázek 22: Obrázek modelu vícepatrového řadového domu s externím výtahem



Obrázek 23: Obrázek modelu vícepatrového rohového řadového domu



Obrázek 24: Obrázek modelu panelového domu

7.2.2. Systém obarvení budov

Pro přidání realismu je při setmění změněn materiál oken. Při této změně je nastavena barva vnitřního osvětlení budovy. Pro různé části budov jsou použity různé barvy. Zde jsou uvedeny jejich hexadecimální kódy, slovní názvy, názvy použití a názvy materiálů v modelu budovy.

Hexadecimální kód barvy	Slovní název barvy	Názvy použití	Název materiálu
A5AED5	Světlá modrá, až tyrkysová	Okna, denní	Material.windowDay
E7E213	Světlá, skoro plně žlutá	Okna, noční	Material.windowNight
E7000D	Světle červená, barva čisté střechy	Střecha	Material.roof

Tabulka 10: Barvy a materiály budov

8. Systém bodování

8.1. Seznam akcí za které se body odečítají

Akce	Počet odečtených bodů
Vykolejení	1000
Projetí zastávky na znamení s cestujícími	500
Převýšení rychlostního limitu	25 za vteřinu
Odjetí kontrolního bodu před dobou odjezdu	250
Neodhlášení zastávky včas (tj. Před odjezdem ze zastávky) nebo špatné odhlášení zastávky	100

Tabulka 11: Seznam akcí za které se body odečítají

8.2. Seznam akcí za které se body přičítají

Akce	Počet přičtených bodů
Zastavení v zastávce	500
Zastavení v zastávce na znamení	1000
Projetí výhybky	200
Odhlášení zastávky	200

Tabulka 12: Seznam akcí za které se body přičítají

8.3. Výpočet výsledné známky

Výsledná známka je určena podílem dosažených bodů a maximálního dosažitelného počtu bodů. Podíl se následně převede na známku dle v programu uložené tabulky. Výchozí nastavení je standardní české známkování od jedné do pěti, kde známka je nejlepší 5 nejhorší. Tabulka, která definuje procentový interval názvy známek je možné upravovat. Maximální počet bodů se vypočítá dle počtu výhybek a zastávek na trati.

Maximální procentuální hranice	Minimální procentuální hranice	Známka
100	86	1
85	66	2
65	41	3
40	21	4
20	0	5

Tabulka 13: Převod procentuálního hodnocení na známky

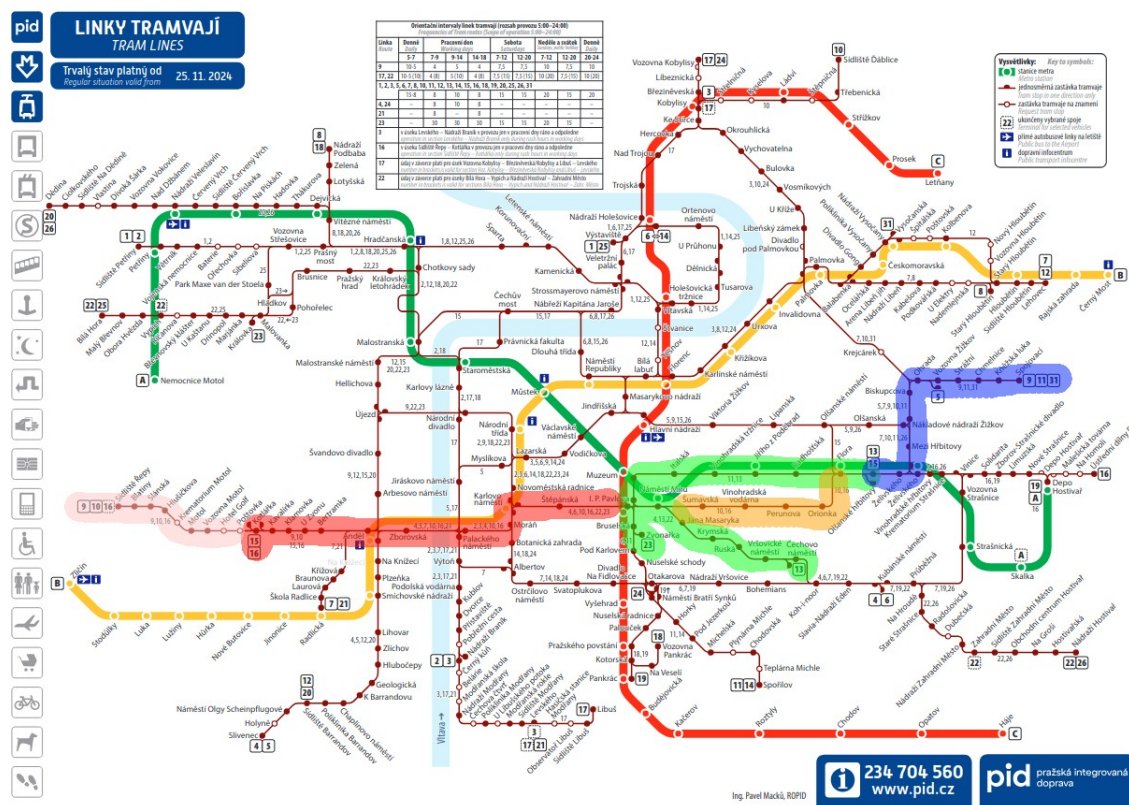
9. Další rozvoj projektu

9.1. Modelování dalších vozidel

Jakékoliv budoucí rozvinutí projektu by potřebovalo přidat i jiné typy tramvajových vozů než pouze typ T3R.P. Nejjednodušší vůz na přidání by byl vůz T3R.PLF, jelikož se jedná o nízkopodlažní variantu s nevelkým počtem změn od původního vozu. Dále by se měl vymodelovat a vložit vůz typu KT8D5.RN2P z důvodů jeho schopnosti obousměrné jízdy. Po přidání tohoto typu by bylo možné vymodelovat další úseky tramvajových tratí, například úsek Levského – Libuš, či simulovat jakoukoliv výluku kdekoli na síti.

9.2. Modelování dalších tramvajových tratí

V budoucnu by měli být přidány další tramvajové tratě navazující na tratě současně v simulátoru obsažené. V první řadě by se tak jednalo o úsek Olšanské hřbitovy – Želivského – Biskupcova – Vozovna Žižkov – Spojovací či úsek Náměstí Míru – Flora přes zastávku Šumavská. Přidáním těchto tratí bude možno simulovat většinu linky 11 (od konečné linky na zastávce Spojovací až do zastávky Bruselská) a značnou část linky 10 (v úseku I. P. Pavlova až Biskupcova). Jako další by se měl implementovat úsek I. P. Pavlova až Kotlářka popřípadě Sídliště Řepy, čímž by byla skoro dokončena linka 9 a další značná část linek 10, 16 a 15. Tento úsek by také povolil demonstraci implementace mostů a řek.



Obrázek 25: Plán přidání dalších tramvajových tratí, vyznačený na mapě PID.^[40]

9.3. Vylepšení modelu osvětlení

Současně používá simulátor model osvětlení Blinn-Phong (viz kapitola 4.6 „Osvětlení“). V budoucnu existuje možnost použití novějšího, avšak pomalejšího, modelu PBR, jinak známého jako Physically Based Rendering (fyzikálně založené vykreslování). Tento model, na rozdíl od modelu Blinn-Phong, zakládá svoje výpočty na fyzikálních jevech či zákonech, nikoliv na několika předem určených barvách a konstantách.

9.4. Přidání řek a mostů

Jakýkoliv další rozvoj projektu by požadoval podporu řek a mostů přes nich. Řeky by bylo možné implementovat nastavením tzv. „úrovně vody“ na kterém by se vykreslovala rovina, popřípadě s shaderem simulujícím efekt vody či ledu. Mosty by bylo potřeba nejdříve celé vymodelovat (a to včetně kolejí, pruhů apod.) a následně napojit na dynamicky vygenerovanou síť tratí. Sloupy mostu by se taktéž museli prodloužit pod úroveň vody (jedná-li se o most přes vodní tok) či pod úroveň terénu, aby nebylo jejich dno vidět. Malé potoky jsou z důvodů časové obtížnosti zanedbány.

9.5. Přidání opravdového terénu

Současně jsou v simulátoru implementovány pouze ulice, koleje a budovy. Pro zvýšení realismu simulátoru by bylo potřeba simulovat i opravdový okolní terén. Jeden způsob kterým by bylo možné tohoto docílit je přidání další vrstvy do scénáře která by určovala převýšení terénu v daném místě. S dostatečnou přesností terénu by bylo možné odebrat současně existující hodnoty výšky pro každý objekt a odvozovat je od hodnot terénu.

9.6. Přidání nastavení trakčního vedení a stahování pantografu

Jelikož na současně simulovaném úseku nejsou žádné mosty pod kterými by tramvaj projížděla a kde by bylo potřeba snížení pantografu, není tato funkce v simulátoru implementována. Současně je trakční vedení vykreslováno na určité výšce nad body tratě a výška pantografu se nemění. Řešením tohoto problému pro budoucí rozvoj projektu je přidání další vrstvy do programu pro úpravy ve kterém by byly přidány výškové body pro trakční vedení, dle kterých by se snižoval a zvyšoval pantograf. Je možné také přidat manuální snížení či zvýšení za pomoci lana v kabině řidiče.

Závěr

Projekt splnil valnou většinu původních očekávání. Obsahuje všechny zmíněné fyzikální simulace, světelnou i zvukovou simulaci, a to vše s velkým počtem optimalizací a na vlastním grafickém rozhraní. Samozřejmě, jak je zmíněno kapitole 9, projekt má ještě dalekou cestu do plného dokončení a plné funkčnosti.

Tento simulátor je již současně možné používat nejen jako nástroj pro trénování budoucí generace řidičů tramvají, ale také jako fyzikální demonstraci jednoduché fyzikální simulace a praktického použití fyziky při výpočtu pohybu kolejového vozidla či jako společenskou demonstraci efektu používání kolejové MHD na emise skleníkových plynů.

Projekt mimo jiné také velice pomohl v rozvoji dovedností týkajících se 3D modelování, konkrétně programu Blender, programování nejen webových ale i desktopových aplikací, fyziky, matematiky, sestavování vlastních formátů a implementace jejich ukládání a načítání hned v několika programovacích jazycích.

Použité technologie

Práce byl zpracována za pomoci následujících technologií:

- Modelování bylo provedeno v softwaru Blender 4.
- Infrastruktura byla implementována za pomoci jazyka C++ a knihoven libc++, GLFW, GLAD, GLM, stbi, fastgltf, Dear ImGui a miniaudio. Bylo taktéž použito API OpenGL 4.5. Pro kompilování programu byl použit systém CMake.
- Webové nástroje byly implementovány za pomoci jazyka JavaScript a technologií HTML a CSS. Byla také použita knihovna Leaflet.js implementující vestavěnou mapu založenou na službě OpenStreetMap (OSM).
- Textur byli vytvořeny v programu Krita 5.
- Úprava zdrojových kódů byla provedena v softwaru KDE Kate pro C++ a konfigurační soubory a v softwaru Microsoft Visual Studio Code pro HTML, CSS a JavaScript.
- Dokument a prezentace byli vytvořeny v programu LibreOffice 24.
- Pro vykreslení grafů byla použita webová aplikace draw.io.
- Jako operační systém byl využit Fedora GNU/Linux 41.

Seznam obrázků

Obrázek 1: Fotografie vozů T3R.P (ev. č. 8234 8235) DPP nedaleko zast. Sídliště Řepy ^[3]	15
Obrázek 2: Červeno-krémový nátěr na modelu tramvaje typu T3R.P.....	15
Obrázek 3: Vůz T3R.PLF (ev. č. 8283) a vlečný vůz T3R.P (ev. č. 8160) na Palmovce ^[3]	15
Obrázek 4: Vínovo-stříbrný nátěr nanesený na modelu typu T3R.P.....	15
Obrázek 5: Umělecké zobrazení nátěru PID vytvořeného dle grafického manuálu PID.....	15
Obrázek 6: Proces vykreslování modelu grafickým rozhraním OpenGL 4.5.....	21
Obrázek 7: Obrázek popisující generaci tratí.....	25
Obrázek 8: Funkce, čísla vzhled jednotlivých tramvajových návěstí. ^{[1][23][24]}	27
Obrázek 9: Možné stavy výhybkových návěstidel tramvajovém simulátoru. ^[1]	29
Obrázek 10: Možné stavy křižovatkových návěstidel tramvajovém simulátoru. ^{[1][25]}	30
Obrázek 11: Fotografie nástroje pro úpravu mapy scénářů při jeho otevření.....	33
Obrázek 12: Fotografie režimu vytváření linek v nástroji pro úpravu mapy scénářů.....	36
Obrázek 13: Fotografie menu nastavení linek v nástroji pro úpravu mapy scénářů.....	36
Obrázek 14: Fotografie nástroje pro vytváření hlásiče při spuštění.....	44
Obrázek 15: Hodnoty použité při výpočtu síly nadzvednutí vozidla v oblouku ^[32]	50
Obrázek 16: Znázornění výpočtu difúzního světla dle (Blinn-)Phongova modelu.....	53
Obrázek 17: Znázornění výpočtu odraženého světla dle Blinn-Phongova modelu.....	53
Obrázek 18: Dopad světla pod úhlem bez aplikování posuvné hodnoty.....	55
Obrázek 19: Dopad světla pod úhlem po aplikování posuvné hodnoty (tzv. „biasu“).	55
Obrázek 20: Obrázek modelu zastávkového sloupu.....	61
Obrázek 21: Obrázek modelu rodinného domku.....	62
Obrázek 22: Obrázek modelu vícepatrového řadového domu s externím výtahem.....	62
Obrázek 23: Obrázek modelu vícepatrového rohového řadového domu.....	63
Obrázek 24: Obrázek modelu panelového domu.....	63
Obrázek 25: Plán přidání dalších tramvajových tratí, vyznačený na mapě PID. ^[40]	67

Seznam tabulek

Tabulka 1: Seznam barev použitých v modelu.....	13
Tabulka 2: Konfigurační hodnoty pro typ T3R.P ^{[13][14]} CITACE ENGINEERINGTOOLBOX	17
Tabulka 3: Posunové hodnoty pro ovládání vozidla typu T3R.P.....	18
Tabulka 4: Další povinné konfigurační hodnoty.....	19
Tabulka 5: První číslice dle vozovny ^{[1][26]}	32
Tabulka 6: Seznam typů budov, jejich popisků a rozměrů.....	40
Tabulka 7: Seznam památek, jejich popisků a rozměrů.....	40
Tabulka 8: Efekt počasí různých stavů na koeficient tření.....	48
Tabulka 9: Hodnoty pro výpočet atenuace závislosti na dosahu světla. ^[20 p. 141]	53
Tabulka 10: Barvy a materiály budov.....	64
Tabulka 11: Seznam akcí za které se body odečítají.....	65
Tabulka 12: Seznam akcí za které se body přičítají.....	65
Tabulka 13: Převod procentuálního hodnocení na známky.....	66

Citace

Je-li citace označena hvězdičkou např. [1*], vztahuje k celému předchozímu paragrafu. Není-li uvedena, vztahuje k předchozí větě, případně několika předchozím (Závisí na kontextu.). Citace jsou uvedeny dle normy ČSN ISO 690.

1. KASTNER, Eduard. ovládání tramvají. Praha, 23. 1. 2025 [osobní komunikace]
2. LEUBNER, Martin. tramvají, nahrávání zvuků na lince 91. Praha, 15. 3. 2025 [osobní komunikace]
3. SEZNAM-AUTOBUSU.CZ. seznam-autobusu.cz. Online. © 2007-2025. Dostupné z: <https://seznam-autobusu.cz/> [cit. 24. 11. 2024]
4. TRANSPORT ÎN COMUN ROMÂNIA. Tramvaie Tatra T3R.P. transport-in-comun.ro. Online. © 2006-2012. Dostupné z: http://transport-in-comun.ro/romania/cehia/praga/tramvaie/t3r_p/t3r_p.html [cit. 24. 11. 2024]
5. DOPRAVNÍ PODNIK HL. M. PRAHY, AKCIOVÁ SPOLEČNOST, ODDĚLENÍ VÝCVIK VZDĚLÁVÁNÍ – TRAMVAJE. Tramvaje T3.SUCS, T3M, T3R.P, T3R.PLF. [Praha], 2009. Dostupné z: https://www.prazsketramvaje.cz/obrazky/stahuj/testy/ucebnice_t3.pdf [cit. 15. 3. 2025]
6. TECHNICKÉ MUZEUM BRNĚ, DEPOZIT VOZIDEL MHD. Brno, tramvaj T3 č. 521. Online. © 2002 Ivo Mahel (obrázek), 2025 (stránka informace). Dostupné z: <https://sabdigital.cz/expozicemhd/exponaty/tramvaje/brno-tramvaj-t3-c-521/> [cit. 14. 9. 2024]
7. TECHNICKÉ MUZEUM BRNĚ, ČKD PRAHA. Technický výkres Pantograf - výkres O-37-170059 – kopie. Online. © Brno [Praha], 2025 [1966]. Dostupné z: <https://sbirkytmb.museion.cz/?id=150091952> Webový portál: Brno, 2025, výkres Praha, 1966. [cit. 14. 9. 2024]
8. KLL-KONSTRUKČNÍ KANCELÁŘ, S.R.O. OSTRAVA. Modernizace tramvajového podvozku t3 sum - pro SKD TRADE, a.s. kll.cz. Online. [© 2008], [Ostrava]. Dostupné z: <http://www.kll.cz/pages/t3sumtramvajpodvozek.html> [cit. 19. 10. 2024]
9. DOPRAVNÍ PODNIK HL. M. PRAHY, A.S. DPP datech. dpp.cz. Online. © 2025. <https://www.dpp.cz/spolecnost/o-spolecnosti/dpp-v-datech> [cit. 26. 3. 2025]
10. MEZINÁRODNÍ ORGANIZACE PRO NORMALIZACI, WIKIPEDIA. wikipedia.org, iso.org. Dostupné z: https://en.wikipedia.org/wiki/ISO_3864 *Tento zdroj byl použit pouze jako zdroj barev použitých specifikaci ISO 3864. Textová informace nebyla použita.* [cit. 24. 11. 2024]

11. WIKIMEDIA COMMONS. Category:ISO 7010 safety signs. commons.wikimedia.org. Online. © 2023. Dostupné z: https://commons.wikimedia.org/wiki/Category:ISO_7010_safety_signs. [cit. 24. 11. 2024]
12. REGIONÁLNÍ ORGANIZÁTOR PRAŽSKÉ INTEGROVANÉ DOPRAVY, INTEGROVANÁ DOPRAVA STŘEDOČESKÉHO KRAJE, PRAŽSKÁ INTEGROVANÁ DOPRAVA. Manuál jednotného vzhledu vozidel PID, autobusy. pid.cz. Online. © 2023. Dostupné z: https://pid.cz/wp-content/uploads/system/sk_bus/SK_bus_priloha_manual_jednotneho_vzhledu_vozidel_2024-01-01.pdf [cit. 24.11.2024]
13. Dopravní podnik hl.m. Prahy, a.s. Tatra T3R.P. dpp.cz. Online. © 2025. Dostupné z: www.dpp.cz/vozovy-park/tramvaje/tatra-t3r-p [cit. 11. 3. 2025]
14. MOBILNÍ TABLA. Informace tramvaji 8228. tram.mobilnidata.cz. Online. © 2022. Dostupné z: <https://tram.mobilnitabla.cz/vuz/8228> [cit. 19. 3. 2025]
15. CPPREFERENCE.COM. C++ reference. en.cppreference.com. Online. © 2025. Dostupné z: <https://en.cppreference.com/w/> [cit. 19. 3. 2025]
16. PETERSEN, Christian. ANSI Escape Sequences. gist.github.com. Online. © 2025. Dostupné z: <https://gist.github.com/fnky/458719343aabd01cfb17a3a4f7296797> [cit. 17. 3. 2025]
17. KHRONOS GROUP, Rendering Pipeline Overview. khronos.org. Online. © 2022. Dostupné z: https://www.khronos.org/opengl/wiki/Rendering_Pipeline_Overview [cit. 17. 3. 2025]
18. KHRONOS GROUP, Interface Block (GLSL): Memory Layout. khronos.org. Online. (c) 2022. Dostupné z: [https://www.khronos.org/opengl/wiki/Interface_Block_\(GLSL\)#Memory_layout](https://www.khronos.org/opengl/wiki/Interface_Block_(GLSL)#Memory_layout) [cit. 17. 3. 2025]
19. KHRONOS GROUP. Shader Compilation. khronos.org. Online. © 2022. Dostupné z: https://www.khronos.org/opengl/wiki/Shader_Compilation [cit. 17. 3. 2025]
20. de VRIES, Joey. Learn OpenGL – Graphics Programming: Learn modern OpenGL graphics programming in step-by-step fashion. 1. vydání. Kendall&Welling. 2020. ISBN 978-90-90-33256-7 Dostupné online ve formátu PDF z: https://learnopengl.com/book/book_pdf.pdf. Zdroj také dostupný ve webové podobě na adrese <https://learnopengl.com>. [cit. 27. 2. 2025]
21. HESHMATI MOGHADDAM, Mahan. LearnOpenGL: OIT: Weighted Blended. learnopengl.com. Online. © 2025. Dostupné <https://learnopengl.com/Guest-Articles/2020/OIT/Weighted-Blended>. [cit. 24.2.2025]
22. DEMES, Lennart. AmbientCG – free textures, HDRIs & More. ambientcg.com. Online. Dostupné z: <https://ambientcg.com/> [cit 15. 3. 2025]

23. PRAŽSKÉ TRAMVAJE. Návěsti návěstidla tramvajové dopravě. prazsketramvaje.cz. Online. © 2001-2025. ISSN 1801-9994. Dostupné z: <https://www.prazsketramvaje.cz/view.php?cisloclanku=2006041260> [cit. 16. 1. 2025]
24. PRAŽSKÉ TRAMVAJE. Návěsti upravující rychlost tramvaje. prazsketramvaje.cz. Online. © 2001-2025. ISSN 1801-9994. Dostupné z: <https://www.prazsketramvaje.cz/view.php?cisloclanku=2006041265> [cit. 16. 1. 2025]
25. PRAŽSKÉ TRAMVAJE. Světelné návěsti pro tramvaje. prazsketramvaje.cz. Online. © 2001-2025. ISSN 1801-9994. Dostupné z: <https://www.prazsketramvaje.cz/view.php?cisloclanku=2006041263> [cit. 16. 1. 2025]
26. CHODOUNSKÝ, Josef. Pořadí na lince. rezistoruv.blogspot.com. Online, blogový příspěvek. © 2019. Dostupné z: <https://rezistoruv.blogspot.com/2020/05/poradi-na-lince.html> [cit. 16. 3. 2025]
27. MOZILLA WEB DOCS. Web technology for developers. developer.mozilla.org. Online. © 2025. Dostupné z: <https://developer.mozilla.org/en-US/docs/Web> [cit. 7. 1. 2025]
28. AGAFONKIN, Volodymyr kol. Leaflet API reference. leafletjs.com. Online. © 2025. Dostupné z: <https://leafletjs.com/reference.html> [cit. 7. 1. 2025]
29. ENGINEERINGTOOLBOX. Rolling Resistance. engineeringtoolbox.com. Online. © 2008. Dostupné z: https://www.engineeringtoolbox.com/rolling-friction-resistance-d_1303.html [cit. 24. 3. 2025]
30. HARMON, M.; SANTA, J.F.; JARAMILLO, J.A. kol. Evaluation of the coefficient of friction of rail in the field and laboratory using several devices. Online. © 2020. Dostupné z: <https://eprints.whiterose.ac.uk/156304/3/Newton%20Friction%20Journal%20Paper%20V2.pdf> [cit. 16. 3. 2025]
31. VOLTR, Petr; MLYNAŘÍK, Ladislav; NOVÁK, Jaroslav; MICHÁLEK, Tomáš. Rozjezd vlaku okem studenta železniční techniky na DFJP. Online, příspěvek. © 2015. Dostupné z: <https://www.zelpage.cz/clanky/rozjezd-vlaku-okem-studenta-zeleznici-techniky-na-dfjp>. [cit. 24. 3. 2025]
32. GHOSH, Shibashis. Tractive Force Calculation for Vehicle. mechguru.com. Online. © 2013. Dostupné z: <https://mechguru.com/vehicle-design/tractive-force-calculation-for-a-vehicle/> [cit. 24. 3. 2025]
33. MARQUIS, Brian; GREIF, Robert. Application of nadal limit in the prediction of wheel climb derailment. In: ASME/IEEE Joint Rail Conference. 2011. p. 273-280. Dostupné z: https://www.railwayage.com/wp-content/uploads/2020/12/JRC2011-56064_nadal.pdf [cit. 16. 3. 2025]

34. ENGINEERINGTOOLBOX. Drag Coefficient. engineeringtoolbox.com. Online. © 2004. Dostupné z: https://www.engineeringtoolbox.com/drag-coefficient-d_627.html [cit 25. 1. 2025]
35. SIDNEY, FMOTION1. What circumstances or material types should have different specular value in PBR workflow? blender.stackexchange.com. Online, příspěvek. © 2021. Dostupné z: <https://blender.stackexchange.com/a/234200> [cit. 15. 12. 2024]
36. UNITED STATES ENVIRONMENTAL PROTECTION AGENCY. Greenhouse Gas Emissions from Typical Passenger Vehicle. epa.gov. Online. © 2024. Dostupné z: <https://www.epa.gov/greenvehicles/greenhouse-gas-emissions-typical-passenger-vehicle> [cit. 24. 3. 2025]
37. EMBER; ENERGY INSTITUTE; OUR WORLD IN DATA. Carbon intensity of electricity generation. ourworldindata.org. Online. © 2024. Dostupné z: <https://ourworldindata.org/grapher/carbon-intensity-electricity> [cit. 24. 3. 2025]
38. GŁÓWNY URZĄD STATYSTYCZNY. Energia 2024 - Folder zawierający podstawowe informacje produkcji wykorzystaniu paliw, ciepła energii elektrycznej 2023 r. Polsce. stat.gov.pl. Online. © 2024. Dostupné z: <https://stat.gov.pl/obszary-tematyczne/srodowisko-energia/energia/energia-2024,1,12.html> [cit. 24. 3. 2025]
39. ČESKÁ NÁRODNÍ BANKA. Kurzy devizového trhu. cnb.cz. Online. © 2025. Dostupné z: https://www.cnb.cz/cs/financni-trhy/devizovy-trh/kurzy-devizoveho-trhu/kurzy-devizoveho-trhu/denni_kurz.txt?date=21.03.2025 [cit. 24. 3. 2025, data 21. 3. 2025]
40. MACKŮ Pavel, PRAŽSKÁ INTEGROVANÁ DOPRAVA. Linky tramvají, trvalý stav platný od 25. 11. 2024. pid.cz. Online. © 2024. Dostupné z: https://pid.cz/wp-content/uploads/mapy/schemata-trvala/a4_tramvajovy_provoz.pdf [cit. 27. 12. 2024]

Následující zdroje byli použity při psaní programu, nejsou avšak samotné dokumentaci práce zmíněny.

41. APELER, Sean. FastglTF Documentation. readthedocs.io. Online. © 2025. Dostupné z: <https://fastgltf.readthedocs.io/v0.8.x/index.html> [cit. 25. 2. 2025]
42. APELER, Sean. gl_viewer.cpp. github.com. Online. © 2025. Dostupné z: https://github.com/spnda/fastgltf/blob/main/examples/gl_viewer/gl_viewer.cpp [cit. 1. 2. 2025]
43. RICCIO, Christophe. OpenGL Mathematics library. Online. © 2025. Dostupné z: <https://glm.g-truc.net/0.9.4/api/> [cit. 22. 2. 2025]
44. VON GAGERN, Martin. Direct way of computing the clockwise angle between two vectors. stackoverflow.com. Online, příspěvek. © 2013. Dostupné z: <https://stackoverflow.com/a/16544330> [cit. 22. 2. 2025]

45. ANONYMNÍ UŽIVATEL [meowgoethedog], How to get right Up vector for LookAt transformation? stackoverflow.com. Online, příspěvek. © 2017. Dostupné z: <https://stackoverflow.com/a/47188758> [cit. 9. 3. 2025]
46. GLFW. Documentation. glfw.org. Online. © 2025. Dostupné z: <https://www.glfw.org/docs/latest/> [cit. 20. 3. 2025]
47. REES, Gareth. How do you detect where two line segments intersect? stackoverflow.com. Online, příspěvek. © 2009. <https://stackoverflow.com/a/565282> [cit. 22. 3. 2025]
48. OPENGL-TUTORIAL. Picking with custom Ray-OBB function. opengl-tutorial.org. Online, blogový příspěvek. © 2013. Dostupné z: <https://www.opengl-tutorial.org/miscellaneous/clicking-on-objects/picking-with-custom-ray-obb-function/> [cit. 15. 3. 2025]
49. OPENGL-TUTORIAL. Billboards: The 3D Way. opengl-tutorial.org. Online, blogový příspěvek. © 2013. Dostupné z: <https://www.opengl-tutorial.org/intermediate-tutorials/billboards-particles/billboards/#solution-2--the-3d-way> [cit. 9. 3. 2025]
50. BITTLE, William. SAT (Separating Axis Theorem). dyn4j.org. Online, blogový příspěvek. © 2010. Dostupné z: <https://dyn4j.org/2010/01/sat/> [cit. 11. 3. 2025]
51. ANONYMNÍ UŽIVATEL [david]. glm translate matrix does not translate the vector. stackoverflow.com. Online, příspěvek. © 2013. Dostupné z: <https://stackoverflow.com/a/18988838> [cit. 13. 3. 2025]
52. BAHANDOV, Dmytro. Ray Oriented bounding box (OBB) intersection function not working with scale. stackoverflow.com. <https://stackoverflow.com/a/76062770> [cit. 16. 3. 2025]
53. REID, David. Miniaudio Programming Manual. miniaud.io. Online. © 2025. Dostupné z: <https://miniaud.io/docs/manual/index.html> [cit. 24. 3. 2025]