

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra kybernetiky

DIPLOMOVÁ PRÁCE

PROHLÁŠENÍ

Předkládám tímto k posouzení a obhajobě diplomovou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

V Plzni dne 15.5.2015

.....

Anotace

Digitalizace dopravních záznamů o provozu vozidel či mechanizačních strojů je v dnešní době nezbytná. Tato diplomová práce se zabývá automatizovaným a inteligentním systémem vytvářející elektronické dopravní záznamy a možností integrace řečových technologií pro syntézu řeči z textu a automatického rozpoznávání řeči. Definuje úlohu, formát dat a přináší informace o základní úspěšnosti využití těchto technologií. Cílem této práce je propojit řečové technologie a inteligentní systém vytvářející dopravní záznamy tak, aby byl použitelný v praxi u dopravních firem.

Klíčová slova

Dopravní záznam, automatizace, stazka, Android, syntéza řeči z text, TTS, automatické rozpoznávání řeči, ASR, IES

Abstract

Digitization of traffic records of transport vehicles operations or mechanized equipment is nowadays essential. This thesis is concerned about automated and intelligent system for generating electronic traffic records and the possibility of integrating speech technologies for speech synthesis and speech recognition. It defines the role, data format and provides information about the success of these technologies. The aim of this work is to connect speech technologies and intelligent system for traffic records to be usable in practice at transport companies.

Keywords

Traffic record, automation, Android, speech synthesis, TTS, automatic speech recognition, ASR, IES

Obsah

| | |
|--|----|
| 1. Úvod..... | 4 |
| 2. Dopravní záznam | 7 |
| 2.1. Kniha jízd a stazka..... | 7 |
| 2.2. Analýza možností řešení | 7 |
| 2.3. Teorie vlastního řešení | 8 |
| 2.4. Průběh vývoje, změny a vylepšení | 8 |
| 2.5. Testování..... | 9 |
| 2.6. Dispečink-řidič-dispečink | 10 |
| 2.6.1. Příklad | 10 |
| 2.7. DMD-IES-DMD..... | 11 |
| 2.8. Automatizace a inteligence..... | 16 |
| 2.8.1. Automatizace | 16 |
| 2.8.1.1. Automatizační jádro | 17 |
| 2.8.1.2. Dočasné procesy | 18 |
| 2.8.1.3. Připojení na sledovací jednotku vozidla..... | 19 |
| 2.8.1.4. Propojení s Bluetooth lokátorem | 21 |
| 2.8.2. Inteligence | 22 |
| 2.8.2.1. Personalizace a učení se..... | 22 |
| 2.8.2.2. Synchronizace dat v reálném čase a IES server | 23 |
| 2.8.2.3. Multimodální interakce | 25 |
| 3. Řečové technologie | 27 |
| 3.1. Syntéza řeči z textu..... | 27 |
| 3.1.1. Korpusové orientovaná syntéza řeči..... | 27 |
| 3.2. Rozpoznávání řeči..... | 28 |
| 3.2.1. Statistické rozpoznávání řeči a skryté Markovovy modely | 29 |

| | |
|---|----|
| 4. Multimodální ovládání Inteligentní Elektronické Stazky | 31 |
| 4.1. Hlasové příkazy a dialogy v Inteligentní Elektronické Stazce | 31 |
| 4.1.1. Globální příkazy pro navigaci a spuštění akce | 32 |
| 4.1.2. Odpovědi na otázky a příkazy v dialozích | 33 |
| 4.1.3. Složitější příkazy zadávání a výběru | 35 |
| 4.2. Implementace syntézy řeči z textu do IES | 36 |
| 4.2.1. Online syntéza řeči na mobilním zařízení | 37 |
| 4.2.2. Offline syntéza řeči na mobilním zařízení | 37 |
| 4.3. Rozpoznávání řeči v IES..... | 38 |
| 4.3.1. Google ASR | 39 |
| 4.3.2. Systém ASR Západočeské univerzity v Plzni | 40 |
| 4.3.3. Implementace ZČU ASR do IES | 41 |
| 4.3.4. Offline systém ASR | 45 |
| 4.4. Shrnutí syntézy řeči z textu a automatického rozpoznávání řeči | 46 |
| 4.5. Schéma využívání TTS a ASR v IES | 47 |
| 5. Aktuální stav, specifikace a budoucnost IES | 49 |
| 6. Závěr | 50 |
| 7. Literatura a reference | 51 |

1. Úvod

V roce 2013 byla ve spolupráci s firmou Hobl&Pech s.r.o [1]. vytvořena bakalářská práce [2], která byla zaměřena na automatizační vrstvu a způsob řešení moderního zaznamenávání dopravních dat. V roce 2014 pak firma Hobl&Pech s.r.o. začala spolupracovat se Západočeskou univerzitou v Plzni [3] na vývojovém projektu Inteligentní Elektronická Stazka [4] pod státním vývojovým programem TA ČR Alfa.

Tato diplomová práce přímo navazuje na bakalářskou práci, dále ji rozvíjí a zkoumá možnosti integrace řečových technologií. Popisuje způsob a průběh vývoje produktu, jeho funkčnost, způsob automatizace, příznaky inteligence a popisuje tak celkový proces moderního vytváření elektronických dopravních záznamů. Následně popisuje možnosti multimediálního ovládání, syntézy řeči z textu a automatického rozpoznávání řeči v operačním systému Android [6]. Při analýze těchto řečových technologií bylo nutné připravit testovací data, navrhnout sady gramatik a otestovat úspěšnost. Následně byly zvoleny nejvhodnější přístupy, které byly implementovány do produktu Inteligentní Elektronická Stazka.

2. Dopravní záznam

2.1. Kniha jízd a stazka

I přes to, jaký je v dnešní době rozmach informačních technologií, stále se kvůli absenci digitalizované podoby dopravních dat nejčastěji používají řidičem ručně psané záznamy dopravních dat. Pro takovýto dokument, obsahující záznamy dopravních dat s vozidlem spojené, se zažil název stazka. Jedna stazka obvykle ohraničuje časový interval od příchodu obsluhy k vozidlu až po jeho odstavení po splnění zakázek zpět do depa. Tento časový interval je v praxi většinou jedna pracovní směna. Jak vypadá takto psaná stazka je ukázáno na obrázku 1.

| VÍCEDEŇNÍ ZÁZNAM O ČINNOSTI NÁKLADNÍHO VOZIDLA/MECHANIZMU PROVOZOVANÉHO PODLE ČL. 3, PÍS. G) NAŘÍZENÍ (ES) Č. 561/2006. ZÁZNAM NAHRADŽUJE POTVRZENÍ PODLE NAŘÍZENÍ RADY (ES) Č. 3821/85ČL. 15 ODSŤ. 2, DRUHÝ PODODST. PÍŠMENO A) ČÍSLO DOKLADU: 106 | | | | PROVOZOVATEL VOZIDLA: HOBL&PECH, S.R.O, NA VRŠÍČKÁCH 7 PLZEŇ | |
|---|---|-----------|------------|--|------------------------------------|
| VOZIDLO/PŘIPOJNÉ VOZIDLO: 4P1 9646 | | K O N E C | | | STÁLÝ ŘIDIČ: Jan Novák |
| K | DEN/ | HODINA | POČÍTADLO | POZNÁMKY (PROSTOJ, ZDRŽENÍ, NÁKUP - TANKOVÁNÍ, POTVRZENÍ, ...) | |
| O | MĚSÍC | | (TACH.MTB) | | |
| D | PŘEVOD Z DOKLADU ČÍSLO: VÝCHOZÍ - CÍLOVÉ MÍSTO JÍZDY, POPIS ČINNOSTI SUBSTRÁT | | | | |
| O | DEPO - PRAHA | 1. 1. | 8:00 | 100 000 | |
| N | DEPO - PRAHA, Palety | 1. 1. | 8:30 | 100 000 | Objednatel Firma 1 |
| J | Firma 1 - BRNO | 1. 1. | 12:00 | 100 250 | |
| V | Firma 1, Palety | 1. 1. | 12:40 | 110 250 | |
| J | DEPO - PRAHA | 1. 1. | 16:30 | 100 500 | Režijní jízda |
| X | DEPO - PRAHA | 1. 1. | 16:40 | 100 500 | |

Obrázek 1 - Ukázka ručně psaného záznamu o provozu vozidla

2.2. Analýza možností řešení

V rámci bakalářské práce byl proveden průzkum aktuálního trhu a možností digitálních záznamů. Bylo zjištěno, že dnes již existují systémy záznamu polohy vozidla díky technologii GPS [5] a také systémy, které umožňují interakci s řidičem. Řidič vozidla či mechanismu tak může zadávat informace o stavu či činnosti vozidla. Tyto systémy se v průzkumu ukázaly jako nevyhovující, jelikož pokud obsluha vozidla informace nezadala, byly informace buď ztraceny, nebo je mohla obsluha libovolně doplnit později a tím byla ztracena věrohodnost daných informací.

2.3. Teorie vlastního řešení

Byla zvolena cesta vývoje vlastního softwaru na telefony a tablety, jelikož jsou v dnešní době tyto zařízení velmi rozšířená a poskytují dostatečné možnosti v řešení tohoto problému. Jako cílový operační systém byl zvolen Android a programátorský jazyk JAVA [36]. V práci byla dále definována teorie datové základny. Tedy následující objekty:

- Okamžitý stav vozidla - Objekt co nejvíce popisující stav vozidla v daném časovém okamžiku.
- Činnost - Objekt na časové linii, popisující skutečnost o režimu či činnosti vozidla v přítomnosti či minulosti.
- Bod zájmu - Objekt popisující známé místo jednoznačně určené svou polohou, názvem a obvyklými činnostmi v místě prováděných.
- Záznam - Kompletní digitální obraz stazky, nebo-li digitální obraz záznamu o provozu a výkonu vozidla či mechanismu v rámci jedné směny.

Dále bylo nutné definovat logická pravidla pro automatizaci zaznamenávání dopravních dat. Byl vytvořen vývojový diagram popisující jeden životní cyklus takzvaného automatizačního jádra. Toto automatizační jádro má za úkol cyklicky zjišťovat aktuální dostupné informace o vozidle a skutečnosti jako takové, tyto informace zaznamenávat a při dostatku informací jednat za obsluhu vozidla. Díky tomu bude pak záznam dopravních dat vytvářen automaticky a obsluha následně doplní pouze doplňující informace neovlivňující důvěryhodnost informací.

2.4. Průběh vývoje, změny a vylepšení

Vývoj započal takzvaně „zcela od nuly“. Bylo třeba vytvořit grafické rozhraní, funkce pro ovládání, výpočty a celkové propojení aplikace. Samotné jádro je část programu, která se v průběhu vývoje měnila nejvíce. Buďto kvůli novým zkušenostem či zlepšení funkčnosti. Jelikož jádro obsahuje veškeré proměnné a objekty, je základ k fungování aplikace, a tudíž bylo vyvíjeno od počátku vývoje. Automatické vlákno jádra však bylo možno vyvíjet až po vytvoření většiny funkcí a možností aplikace.

V průběhu vývoje se měnily přístupy k různým objektům či hardwaru, byly testovány různé modely blokových diagramů a také způsoby programátorských pohledů na jádro. Od druhé generace již má jádro programátorský model neměnný a mění se pouze algoritmus jako takový.

Na začátku vývoje obsluhovalo vlákno jádra krom vytváření nynějšího stavu pouze situace, kdy nebylo jasné, jaká je nynější činnost vozidla či řidiče. V průběhu vývoje bylo samozřejmě nasazeno mnoho vylepšení. Tím je například možnost hlídání bezpečnostních přestávek, možnost připojit se přes technologii Bluetooth [7] k jednotce vozidla, mapování ujeté cesty či kontrola, zda má řidič dostatek času na plánované činnosti.

2.5. Testování

Jsme v situaci, kdy jsme vytvořili nový algoritmus jádra a chceme ho otestovat. Ať už je účel algoritmu jakýkoli, je spuštěn při nějaké situaci, a to buď programové či reálné. Pokud chceme správnost algoritmu otestovat, musíme zkušební situaci buďto nasimulovat, či reálně vytvořit. Při testování aplikace bylo použito simulace ve virtuálním prostředí, simulace v praxi a testování v reálném provozu.

Simulaci ve virtuálním prostředí situací lze provést již na PC díky emulátoru zařízení s Androidem. Takovou simulací snadno ověříme správnost výpočtů, vnitřních vazeb aplikace či grafického rozhraní. Máme i možnost simulace příjmu GPS souřadnic, která je však omezená. Pokud simulace proběhne bez problémů, je pravděpodobné, že programátorský kód je napsán správně a můžeme přejít na simulaci v praxi.

Simulace v praxi pro nás znamená, že se zařízením, kde je v chodu naše aplikace, budeme reálně pohybovat vozidlem a provádět napodobení různých činností či situací. Tím lze z části ověřit správnost modelů chování, správnost rozhodování jádra a další. Během testování v této fázi již zaznamenáváme digitální záznam simulované jízdy a na PC tak můžeme dále ověřit správnost zápisu záznamu. Pokud i tato simulace proběhne bez problémů, lze přejít na testování algoritmu v reálném provozu.

Testování v reálném provozu je pojem označující reálnou jízdu s reálnými činnostmi, bez jakékoli simulace, kdy je nutné domluvit reálnou zakázku s dopravním vozidlem

a oprávněnou osobou a provádět normální pracovní činnost. Již se nesnažíme dostat aplikaci do konkrétní situace, ale testujeme aplikaci jako celek a kontrolujeme správné chování aplikace i automatizačního jádra. Při tomto testování můžeme simulovat pouze lidské chyby během interakce s aplikací. Získáme tím maximální množství informací o správnosti algoritmu a funkčnosti celé aplikace. Po ukončení reálného testování již máme plnohodnotný digitální záznam stazky [2].

2.6. Dispečink-řidič-dispečink

Proces vytváření digitálního záznamu dopravních dat není spojen pouze se samotným sběrem dat, ale i s dispečinkem firmy, který informace nejen kontroluje, ale i zadává do interního systému. V této kapitole se zaměříme na celý proces jedné směny nákladního vozidla a s ním spojené data.

2.6.1. Příklad

Dopravní firma dostane zakázku na dovoz kamene z lomu na stavenišťě. Odpovědná osoba zakázku přijme a vloží ji do počítačového programu pro dispečink. Dispečer zakázku převezme, vyhodnotí pro ni vhodné vozidlo, čas provedení, místo nakládky a vykládky, či další. Zakázku přiřadí k vozidlu, k němuž také přiřadí vhodného řidiče. Tím dispečer vytvoří plán činností pro konkrétního řidiče a vozidlo na daný čas.

Řidič začne směnu a převezme si plán na dnešní směnu. Plán akceptuje a převezme dané vozidlo. S vozidlem pak plní plánované činnosti co nejlépe může. V tomto případě jede do předem určeného lomu kamení. Po nakládce pokračuje další plánovanou činností, přejezdem na stavenišťě a následnou vykládkou. Poté se vrací zpět na firmu, kde odstaví vozidlo, předá splněný plán a záznam o provozu vozidla zpět dispečerovi.

Dispečer plán zkontroluje a označí jej za splněný. Poté je možné vytvořit fakturu pro zákazníka zadávajícího zakázku dle přesných údajů, tedy cena převozu dle ujetých kilometrů plus cena materiálu. Dále je možné vypočítat výplatu řidiči vozidla za reálně odvedenou práci. Vedoucí dále může zkontrolovat kvalitu práce dispečerů, plnění zakázek a další. To vše díky správným informacím.

Tento teoretický příklad je sice ideální, nicméně v realitě velmi obtížně proveditelný a to hned z několik důvodů:

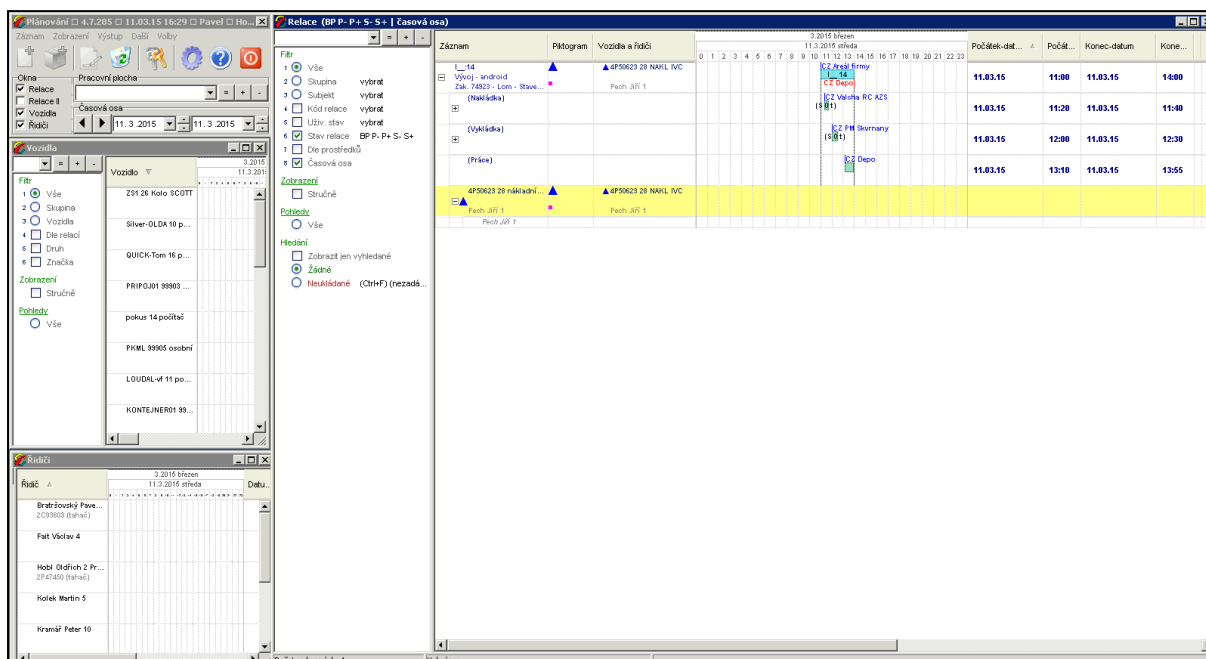
- Dispečink musí disponovat sofistikovaným programem pro plánování.
- Plán musí být detailně naplánován.
- Řidič musí svědomitě zaznamenávat informace o realitě v přesných časových okamžicích.
- Řidič se musí pevně držet plánu.
- Záznam o provozu vozidla musí být co nejpřesnější.

Dále pak musí být vyřešeny situace např. pokud bychom chtěli změnit plán u vozidla, které již plán plní, či např. pokud by v daném lomu nebyl požadovaný substrát. Dále bude představeno kompletní řešení pro celý tento proces .

2.7. DMD-IES-DMD

Firma Hobl&Pech s.r.o. stojí za vývojem nejen Inteligentní Elektronické Stazky (zkratka IES), ale i softwarem DMD [8], nebo-li dispečinkem pro dopravní a převozní firmy. Popis tohoto software je nad rámec diplomové práce, ale bude ukázána práce s ním, pro plnou demonstraci celého procesu zakázky pro dopravní firmu.

Dispečer v plánovacím režimu v softwaru DMD vytvoří plán pro danou zakázku, přiřadí k ní potřebné údaje o objednateli, časových dispozicích a další. Dále pro zakázku vytvoří jednotlivé plánované činnosti, které může specifikovat dopodrobna. Poté má k dispozici seznam všech řidičů a vozidel, u kterých je poskytnuta informace o využití, a zda-li jsou v daném časovém rámci k dispozici. Příklad takové plánované zakázky je vidět na obrázku 2.



Obrázek 2 - Ukázka plánování zakázek v programu DMD

Jakmile je zakázka připravena, je automaticky exportována na FTP [9] server. V této chvíli je dispečer s plánováním hotov a přichází na řadu řidič vozidla. Na tabletu řidič spustí IES, která se automaticky synchronizuje s FTP serverem. Řidič tak má možnost shlédnout své plány pro dnešní den. Poté zvolí konkrétní vozidlo a potvrdí přijetí plánu pro dané vozidlo, tím začíná stazka. IES je rozdělena do několika obrazovek. Obrazovka souhrn slouží k zobrazení všech důležitých informací na jednom místě. Těmi jsou poloha, aktuální režim, informace o bezpečnostních přestávkách, informace o aktivních zakázkách a dosud neprovedené úkoly v rámci zakázek. Dále jsou zde možnosti pro ruční změnu režimu a další. Obrazovka Zakázky slouží k podrobným informacím o zakázkách k danému vozidlu a řidiči. Zakázky mohou být aktivní či neaktivní, dokud je zakázka aktivní, budou provedené činnosti účtovány objednateli zakázky. Obrazovka Vozidlo pak zobrazuje, popřípadě umožňuje doplnit aktuální informace o vozidle. Tyto obrazovky jsou zobrazeny na obrázku 3.



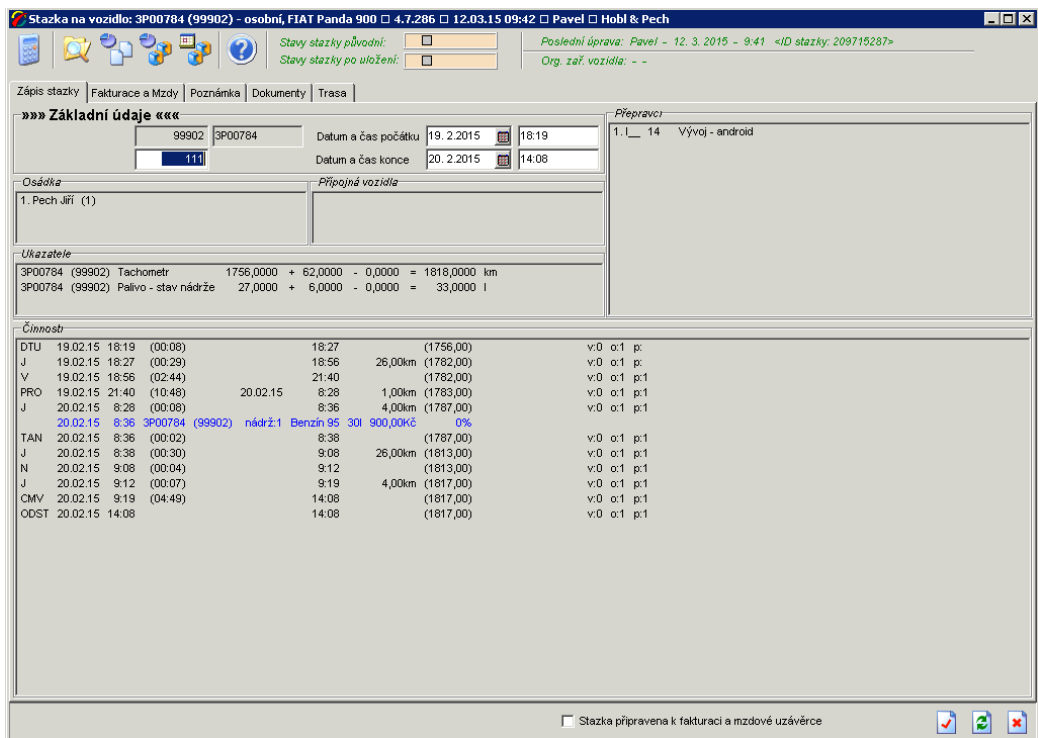
Obrázek 3 - Ukázka záložek souhrn, zakázky a vozidlo z aplikace IES

IES během směny zaznamenává průběh stazky a to buď zcela automaticky, či s pomocí řidiče. Řidič má kdykoli v průběhu směny možnost zkontrolovat dosud vytvořený záznam stazky na obrazovce Stazka. V záznamu může či musí řidič doplnit potřebné informace pro kompletnost stazky. Na konci směny musí řidič zkontrolovat a odsouhlasit koncové informace o vozidle. Poté doplnit informace do záznamu stazky a také ji odsouhlasit. Poté IES opět provede synchronizaci s FTP serverem a skončí. Příklad úpravy stazky na konci směny je na obrázku 4.

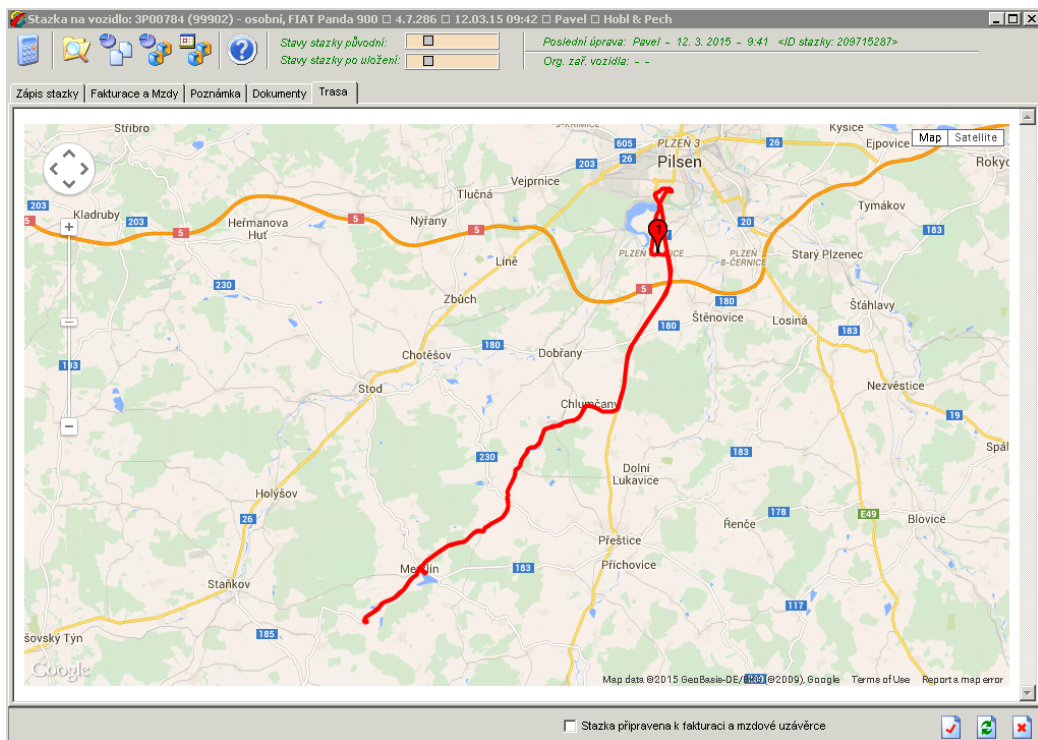


Obrázek 4 - Ukázka záložky stazka z aplikace IES

Vrátíme se zpět do DMD, kde se automaticky importovala stazka z FTP serveru. Po jejím načtení lze vidět průběh stazky, jednotlivé činnosti a to ve formě ideální pro dispečera. Dále je zde možnost přímo vypočítat fakturace a mzdy za danou stazku. Samozřejmostí je i zobrazení záznamu jízdy na mapových podkladech. Příklad výsledné stazky v DMD je na obrázku 5 a její záznam jízdy na mapových podkladech je na obrázku 6.



Obrázek 5 - Ukázka zobrazení vytvořené stazky v programu DMD



Obrázek 6 - Ukázka zobrazení provedené stazky na mapových podkladech

2.8. Automatizace a inteligence

V této kapitole se zaměříme na význam automatizace a inteligence v rámci Inteligentní Elektronické Stazky. Automatizaci lze obecně chápat jako minimalizaci potřeby přítomnosti člověka při vykonávání určité činnosti. V našem případě má automatizace význam v automatickém jednání takovém, které by vykonávala obsluha. Tím minimalizujeme potřebu interakce řidiče vozidla a IES. Inteligence je naopak schopnost řešit nově vzniklé nebo obtížné situace, učit se ze zkušeností či se přizpůsobit novým okolnostem. Dále bude podrobně vysvětlena automatizace a inteligence IES.

2.8.1. Automatizace

Snaha automatizovat jednání byla již od počátku vývoje Elektronické stazky. Automatizovaných procesů přibývalo s vývojem a testováním. Hlavním prvkem v automatizaci je automatizační jádro, které, jak bylo řečené výše, bylo popsáno v bakalářské práci. Automatizační jádro však v průběhu dalšího vývoje začalo být nedostačující, jelikož nebylo vhodné, aby obstarávalo paralelní procesy v rámci celé aplikace. Proto bylo při vývoji IES přistoupeno nejen k rozšíření působnosti automatizačního jádra, ale bylo také implementováno mnoho automatizačních procesů, které mohou pracovat v asynchronním režimu s automatizačním jádrem. Na rozdíl od automatizačního jádra, které cyklicky provádí svůj životní cyklus, po celou dobu běhu IES se tyto procesy spouští pouze v konkrétních situacích, kdy je nutno jednat za obsluhu a po vyřešení těchto situací tyto procesy končí. Pouze dočasná potřeba těchto procesů byla hlavní důvod pro separaci jádra a jednotlivých procesů.

Příkladem činnosti jádra může být cyklické ověřování aktuální polohy a výpočty, zda-li se nacházíme uvnitř nějakého bodu zájmu. Tato informace je potřebná v rámci celé aplikace a je nutné, aby byla aktuální po celý její běh. Naopak dočasným procesem je například situace, kdy je při započetí nového režimu nutné založit nový bod zájmu, avšak informace o poloze nejsou dostatečně přesné. Jádro potřebuje pracovat s aktuálními informacemi, proto nesmí nepřesné údaje zahazovat a také nemůže čekat na údaje přesné. Proto je v této situaci spuštěn proces, který umožní započetí nového režimu a práci jak jádra, tak IES a přitom v pozadí vyčkává na přesné informace o poloze, které následně doplní do již existujících objektů.

Dále budou ukázány některé funkce automatizačního jádra a samostatných procesů. Vzhledem k tomu, že konkrétní logické schéma chování je chráněno autorským právem, budou tyto situace popsány spíše teoreticky.

2.8.1.1. Automatizační jádro

Automatizační jádro provádí svůj životní cyklus cyklicky a to každou jednu sekundu. Zaměříme se na situace, kdy má automatizace za úkol nahradit chování obsluhy:

- Ukončení nynějšího režimu vozidla - Pokud je vozidlo v jiném režimu než jízda, například v režimu nakládky, je jisté, že tento režim probíhá v rámci bodu zájmu. Pokud obsluha nezvolí ukončení režimu ručně, musí režim ukončit jádro samostatně. Pokud vozidlo opustí bod zájmu a pokračuje v pohybu, je jisté, že přešlo do režimu jízda a jádro nakládku automaticky ukončí. V takových situacích se ale musí počítat s případy, kdy jsou k dispozici pouze nepřesné údaje o poloze a rychlosti, které mohou vypovídat o pohybu mimo bod zájmu ale reálně přitom může být vozidlo nehybné. Dále je nutné počítat například s tím, že bod zájmu je chybně definován dispečerem a nakládka může započít uvnitř a pokračovat vně bodu zájmu. Ukončení nynějšího režimu spojeného s daným místem může způsobit i plánovaná zakázka, a to v případě, kdy jsou v plánu dva režimy jdoucí za sebou na stejném místě. Pokud v takové situaci časy plánovaných režimů odpovídají realitě a řidič neodpovídá na dotaz IES, je pravděpodobnější, že aktuální režim přešel do druhého plánovaného. Pokud bychom tak neprovedli, mohlo by se stát, že by řidič bez interakce s IES odjel z bodu zájmu a v záznamu stazky by nebyla obsažena druhá plánovaná činnost. Speciálním případem ukončení nynějšího režimu je režim jízda, který za určitých podmínek končí s pohybem vozidla.
- Započítí režimu - Výhodným faktem je skutečnost, že pokud není vozidlo v pohybu, není režim jízda a je vozidlo v nějakém jiném režimu. Druhy režimů se liší, ať už podstatou vozidla samotného, či požadavky zákazníka využívající IES. Mezi základní ale patří nakládka, vykládka, prostoj, mechanizační činnost, tankování, příprava před jízdou, odstavení vozidla, soukromá jízda a činnost mimo vozidlo. Pro některé režimy je implementováno specifické chování IES. Tyto režimy mohou začínat za různých podmínek a situací. Dvěma hlavními skupinami jsou situace, kdy jízda skončila uvnitř

či vně bodu zájmu. Příkladem může být, že v jedné situaci zastavilo vozidlo v areálu dodavatele produktů, který je ohraničen bodem zájmu, u kterého je jako pravděpodobná činnost definovaná nakládka produktů. V takové situaci je velmi pravděpodobné, že začala nakládka. Pokud je navíc režim plánován a je časové rozmezí přijatelné, je začátek nakládky téměř jistý. Naopak pokud vozidlo ukončí režim jízda na zcela neznámém místě, mnoho informací k dispozici není. Pokud se do nějaké doby vozidlo dá opět do pohybu, je pravděpodobné, že šlo pouze o zastavení z důvodu dopravní situace. Pokud je však vozidlo nehybné delší dobu, je možné, že začal prostoj, bezpečnostní přestávka, či zcela jiný režim. Také je nutné počítat se situací, kdy dispečer dopředu nezná přesnou lokaci plánovaného režimu a definuje tak zatím neurčitý bod zájmu s přiřazenou činností. IES pak musí tento režim rozpoznat, dodefinovat a následně zajistit, aby se aktualizoval v databázi dispečera.

2.8.1.2. Dočasné procesy

- Zjištění aktuálního místa - Tento dočasný proces je spuštěn při započatí popřípadě ukončení režimu. Má za úkol udělat vše pro to, aby řidič nemusel vyplňovat název místa s režimem spojeného. V situaci, kdy je vozidlo v bodu zájmu a informace jsou přesné, je tento úkol vcelku jednoduchý. V mnoha případech se však stává, že jsou buď aktuální informace o poloze nepřesné, či poloha vozidla není zcela v bodu zájmu. Příkladem může být, že v bodě zájmu, kde je plánován určitý režim, není místo k zaparkování a řidič následně zastaví poblíž bodu zájmu. V takové situaci má tento proces za úkol zjišťovat informace po delší časový úsek a následně rozhodnout, zda-li blízká vzdálenost k bodu zájmu znamená jeho přiřazení bodu zájmu k režimu či k režimu přiřadit zcela nový bod zájmu. Také je nutné brát v potaz, zda-li režim začal ručním zadáním řidiče, či byl započat automaticky. Pokud byl totiž započat automaticky z důvodu nereagování obsluhy, je dost pravděpodobné, že ani na další dotazy v tuto chvíli obsluha reagovat nebude.
- Vytvoření nového bodu zájmu - Tento proces již byl naznačen výše a má za úkol zakládat co nejpřesněji nové body zájmu ve chvílích, kdy nejsou na začátku režimu informace o poloze dostatečně přesné. Tento proces může běžet v průběhu celého běhu režimu, jelikož má snahu získat co nejvíce informací o novém bodu zájmu tak, aby byl co nejpřesněji určen pro další zpracování dispečerem.

- Automatické jednání v dialogích - Aplikace obsahuje několik dialogových oken s dotazy, možnostmi doplnění údajů a další. Je nutné ošetřit i situace, kdy řidič například zvolí možnost započít novó režim, avšak dále již nereaguje. Pokud by bylo spuštěno jen jádro, sice by byl automaticky spuštěn novó režim, avšak v tomto dialogu jsou jiné podmínky pro spuštění, jelikož je jisté, že novó režim začíná. U několika důležitých dialogů jsou implementovány procesy, které při nečinnosti obsluhy jednají v rámci dialogu za obsluhu.
- Výpadek a ověření GPS - Další dočasným procesem je takový, který se spustí při celkovém výpadku GPS lokátoru. Takový výpadek může být způsoben vjezdem do tunelu, či špatným signálem. V takových situacích musí IES stále pracovat, avšak nesmí tento výpadek v případě předchozího pohybu znamenat, že pohyb přestal. Byl by tak totiž ukončen režim jízda. IES následně spustí proces, který prohledává stavy vozidla, polohy a rychlosti do minulosti a podle nich vytváří imaginární stavy takové, aby byla IES co nejméně ovlivněna výpadkem GPS do doby, než je lokace obnovena. Musí se však také počítat s případy, kdy například vjede nákladní vozidlo do lesa, kde bude kvůli špatnému signálu výpadek signálu a vozidlo začne nakládku dřeva. V takovém případě naopak tento proces musí zajistit ukončení režimu jízda a spuštění režimu nového.

2.8.1.3. Připojení na sledovací jednotku vozidla

Mobilní zařízení, na kterém běží IES má omezenou množinu informací, kterou je schopna zjistit. Například je nemožné bez dalších periférií zjistit, zda-li je mobilní zařízení uvnitř vozidla, zda-li je motor vozidla v chodu, či další podstatné informace o vozidle. V průběhu vývoje firma Hobl&Pech začala spolupracovat s firmou Princip a.s. [10], která vyvíjí monitorovací jednotku Vectronics, a firmou GX SOLUTIONS BOHEMIA s. r.o. [11], která vyvíjí jednotku Torola Q. Obě jednotky se vyznačují následujícími vlastnostmi:

- Vlastní přijímač GPS souřadnic.
- Sběr informací ze sběrnicí CAN [12] nebo FMS [13]. Oba tyto typy sběrnic jsou často využívány v automobilové diagnostice.
- Možnost analogových vstupů u starší vozidel.

- Komunikace přes Bluetooth 2.0 nebo Bluetooth 4.0 [14].
- Možnost GPRS [15] komunikace pomocí SIM karty.
- Možnost získat mnoho relevantních informací o vozidle v reálném čase.

Obě tyto jednotky byly instalovány do firemního testovacího vozidla IVECO EuroCargo 100 E. Toto vozidlo je v praxi běžně využíváno k přepravě materiálu či mechanickým pracím a je proto pro testování ideální. Dále byl pro IES vyvinut modul pro Bluetooth komunikaci s těmito sledovacími jednotkami. Jelikož jednotka Vectronics komunikuje pouze přes novější rozhraní Bluetooth 4.0 (zkratka BLE), je komunikace s každou jednotkou rozdílná. Pokud však nebudeme brát v potaz programátorskou implementaci, je využití jednotek shodné.

Jakmile řidič začne stazku na konkrétním vozidle, IES se začne automaticky připojovat na monitorovací jednotku vozidla. Po úspěšném připojení již cyklicky čte data z jednotky a převádí je na čitelný formát. Z jednotek lze vyčíst mnoho užitečných informací:

- Ověření GPS souřadnic jednotky a mobilního zařízení. Tím se razantně zmenší nepřesnost těchto souřadnic.
- Získání informace o reálné rychlosti vozidla odečtená z tachometru. Taková informace je věrohodnější než rychlost vypočtená z GPS souřadnic.
- Získání informací o stavech ukazatelů, konkrétně stav počítadla ujetých kilometrů, stav paliva v nádrži a stav motohodin.
- Pravdivostní informace typu motor zapnut, hydraulická ruka v chodu, sklápěcí mechanismus v chodu a další.
- Informace o řidiči či spolujezdci z identifikačních karet a jejich informace o dodržování AETR [16].
- Mnoho dalších možných informací ze sběrnic CAN a FMS, například spotřeba paliva, otáčky motoru, či další.

Testování jednotek a Bluetooth modulu probíhalo při reálných jízdách, aby byla platnost informací jistá. Obě jednotky se chovají velmi podobně jen s mírnými odlišnostmi. V praxi se ukázalo, že využívání IES se sledovací jednotkou značně snižuje chybovost vytvořené stazky, řidič nemusí vyplňovat tolik informací a celkové logické chování IES je spolehlivější a rychlejší. Například pokud vozidlo dlouho stojí na neznámém místě a motor je zapnut, je velmi pravděpodobné, že vozidlo stojí v koloně aut, na semaforu či na přejezdu.

Pokud ale vozidlo zastaví na neznámém místě, vypne motor a zapne hydraulickou ruku, je jisté, že skončil režim jízda a začíná jiný režim, například mechanizační činnost. Bez sledovací jednotky má IES informaci pouze o tom, že vozidlo stojí nějaký čas na neznámém místě a musí tak jednat velmi opatrně a vyžaduje větší kontrolu od řidiče.

2.8.1.4. Propojení s Bluetooth lokátorem

Během testování IES u zákazníků bylo zjištěno, že i přes skutečnost, že se sledovací jednotkou funguje IES mnohem lépe, někteří zákazníci instalaci monitorovacích jednotek do svých vozidel nechtějí. Hlavním důvodem je cena jednotky a její instalace, která převyšuje cenu IES. Firma Hobl&Pech proto přišla s kompromisem mezi sledovací jednotkou a využíváním IES bez jednotky. Ve spolupráci s firmou Princip a.s. započal vývoj jednoduchých Bluetooth lokátorů, které budou napevno umístěny ve vozidle. Tento lokátor není větší než běžné hodinky, funguje na baterii a chová se jako Bluetooth vysílač. IES je schopna, pokud je lokátor v blízkosti mobilního zařízení, se na tento lokátor připojit. Jakmile je mobilní zařízení mimo dosah lokátoru, spojení je přerušeno. Tato jednoduchá informace má pro IES zásadní význam, jelikož lze takto zjistit, zda-li je mobilní zařízení ve vozidle či nikoliv. Tato informace je velmi podstatná v několika situacích, například:

- Ztráta GPS signálu - Pokud je GPS signál ztracen či je výrazně snížena jeho přesnost, je tento jev většinou doprovázen předchozím tzv. „uletěním“ signálu. To se dá vysvětlit tak, že signál postupně ztrácí svou přesnost, tedy zvětšuje se vzdálenost mezi získanými a reálnými souřadnicemi. Tento jev se často jeví jako jízda, i když reálně vozidlo stojí nehybně namístě. Ztráta GPS signálu se však reálně děje výjimečně, zpravidla při vjezdu do tunelu nebo vchodu do budovy s mobilním zařízením. V případě, kdy řidič započne jiný režim než jízda a vejde do budovy s mobilním zařízením u sebe, lze ztrátu GPS signálu chápat jako přechod do režimu jízda, i když skutečnost je jiná. Pokud bychom pomocí lokátoru v těchto situacích věděli, zda-li je mobilní zařízení uvnitř vozidla či nikoliv, mohli bychom tyto situace vyřešit zahazením nepřesných informací, pokud není zařízení ve vozidle.
- Odchod z bodu zájmu - IES umožňuje režim *Činnost mimo vozidlo*, který je zde pro situace, kdy řidič potřebuje vystoupit z vozidla a pokračovat ve stazce chůzí. Poté se opět vrátí do vozidla a pokračuje ve stazce s vozidlem. Tento režim je speciální v tom,

že IES si nikdy nemůže být jistá, zda-li řidič odjíždí z daného místa pomalou jízdou ve vozidle způsobenou dopravní situací, či zda-li opouští místo chůzí. Při vývoji byla vytvořena logika chování v tomto režimu, avšak nikdy není zcela spolehlivá, jelikož v průběhu režimu mimo vozidlo může řidič několikrát projít daným místem a opět ho opustit. Při využití lokátoru tento problém zcela odpadá a navíc lze tuto logiku využít i ve všech ostatních režimech.

Vývoj těchto lokátorů je v testovací fázi, kdy již testujeme prototypy zařízení a vyvíjíme logiku chování pro IES. Cena lokátoru je oproti jednotce výrazně malá, lokátor nepotřebuje žádnou instalaci do vozidla a na baterii vydrží v chodu 1 až 2 roky.

2.8.2. Intelligence

Jak již bylo řečeno, intelligence je schopnost řešit nové, neznámé situace, popřípadě schopnost učit se. V rámci projektu IES však budeme za projev intelligence považovat i natolik chytré jednání, u kterého se již nedá říci, že je to čistá automatizace.

2.8.2.1. Personalizace a učení se

Z předchozího již víme, že IES je schopna zakládat nové body zájmu. Zde však proces nekončí. Při založení nového bodu zájmu je tento bod automaticky odeslán do dispečinku. Zde pověřený pracovník daný bod upraví či zkontroluje a poté je tento bod zařazen do katalogu všech bodů zájmu firmy. Každý bod je však nejen unikátní svou polohou ale i činnostmi v něm prováděných. IES při každém provedení režimu zapíše tento režim do vlastností bodu zájmu a tato informace se dostane do katalogu firmy. Pokud následně přijede jiný řidič do stejného místa, IES má okamžitě informaci o pravděpodobném režimu, kterou dále využije pokud řidič nereaguje. Pokud například byla provedena na nějakém místě několikrát nakládka a to i různými řidiči, je velmi pravděpodobné, že pokud řidič nezvolí jinak, je opět prováděna nakládka. Touto schopností učit se vlastnosti bodů zájmu jsme schopni blíže určit skutečnost v případě nedostatku dalších informací.

Dalším důležitým faktorem je personalizace vozidel a chování IES. Mezi personalizaci samotné IES můžeme zařadit volbu některých konstant, úpravy chování v některých

situacích, práci IES se zdroji zařízení či způsob nahlížení na některé režimy. Personalizace vozidel je důležitá v tom, že je možné nastavit různé atributy různým vozidlům. Například vozidlo se sklápěcím mechanismem je ze své podstaty schopné provádět jiné režimy, než vozidlo přepravního typu. Proto jsou u různých vozidel zpřístupněny jen určité režimy vozidla. Dále je vzhledem k vozidlům přizpůsobeno samotné chování IES. Příkladem může být situace, kdy jízda se zapnutým hydraulickým čerpadlem je u vozidla se sklápěcím mechanismem nevhodná a nebezpečná, avšak u vozidla se sypacím mechanismem normální.

2.8.2.2. Synchronizace dat v reálném čase a IES server

Jak bylo popsáno výše, způsob předávání dat byl u ES proveden přes FTP server. ES data předává na začátku a na konci stazky, popřípadě stahuje data při aktualizaci plánů. Z pohledu dispečinku musí dispečer ručně spustit import provedených stazek z FTP serveru. Toto řešení bylo pro IES nevhodné a musel být proto vymyšlen nový způsob předávání dat a to nejlépe v reálném čase.

Prvotní myšlenkou bylo vytvořit server, který by obstarával synchronizaci se všemi zařízeními s IES, byl zcela samostatný a byl přístupný i pro více dispečinků. Po důkladné analýze byl jako hlavní stavební prvek vybrán software Couchbase Server [19]. Tento software obstarává databázi JSON [17] dokumentů, je velmi robustní a upravitelný. Couchbase databáze vyniká nejen v tom, že je tzv. noSQL [18], nebo-li že nefunguje na klasickém SQL jazyku ale i tím, že své dokumenty uchovává převážně v paměti RAM a její přístup a pohledy na databázi jsou tak velmi rychlé. To se velmi hodí při velkém objemu dat.

Couchbase Lite [19] je upravená verze Couchbase databáze na různé platformy. V případě IES je důležitá platforma Android. S tímto softwarem jsme schopni v zařízení s IES obsluhovat lokální databázi pracující opět s JSON objekty. Na tuto lokální databázi je také možné dělat různé pohledy jako v případě Couchbase Server. Tyto prvky by však nebyly užitečné bez následujícího velmi důležitého prvku, kterým je Couchbase SyncGateway [19]. Tento software je synchronizační brána mezi serverem a zařízením s Androidem.

Při správném nastavení lze docílit díky synchronizační bráně toho, aby se automaticky synchronizovala data mezi mobilními zařízeními a serverem. Tato synchronizace probíhá jako služba na pozadí a to i při vypnuté IES. Pokud je zařízení připojené k internetu, jsou data na

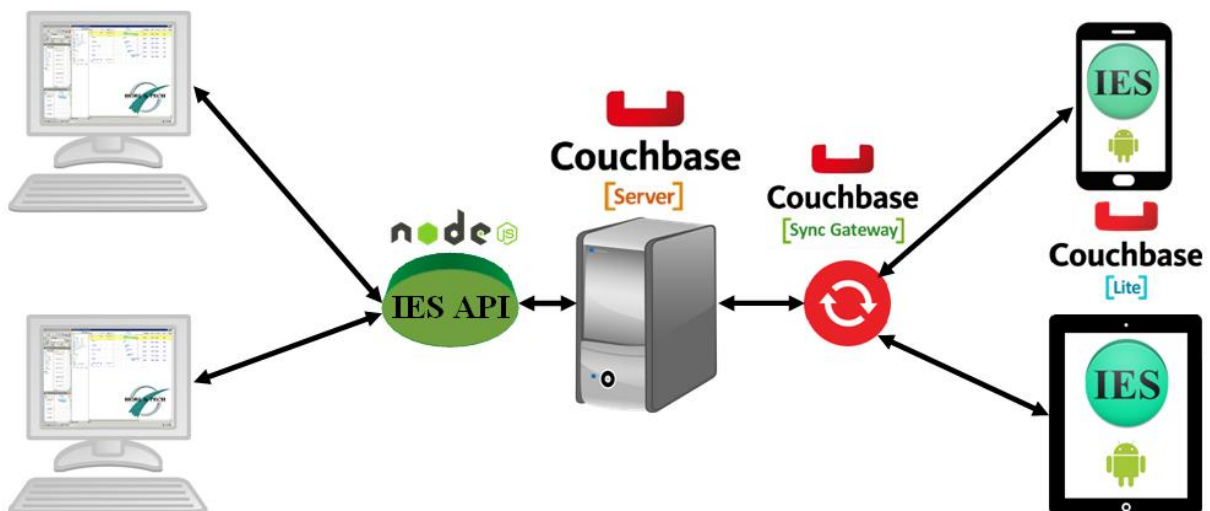
serveru aktuální v reálném čase. Podstatné je i to, že lokální databáze na mobilním zařízení může být pouze částečný klon databáze na serveru. Lze tak docílit toho, aby v lokální databázi byly jen dokumenty spojené s daným zařízením a řidiči, jež toto zařízení využívají. Tím zabezpečíme nejen zbytečný přenos dat, ale i to, aby na zařízení byla jen data určená pro toto zařízení.

V předešlém je vyřešena synchronizace mezi serverem a mobilními zařízeními. Nyní se zaměříme na synchronizaci mezi dispečinkem a serverem. Byla uvažována možnost, že by server běžel na stejném počítači jako dispečink, nicméně tato možnost by nás ochudila o možné výhody samostatného serveru. Proto byla vytvořena koncepce IES serveru. Na tomto serveru je nutné rozhraní pro práci s databází ze vzdáleného počítače. Pro toto API rozhraní byl zvolen software Node.js, který kromě jiného umožňuje napojení na Couchbase Server.

Na API rozhraní k IES serveru byly požadavky, aby bylo možné po zabezpečené lince vytvářet a upravovat data v databázi a aby bylo možné co nejrychleji získávat určité dokumenty dle pohledů na databázi. Navíc by tato přenášená data měla být ve formátu vhodném pro dispečink. V softwaru Node.js byl vytvořen program, který je napojen na Couchbase server, obstarává HTTP [20] dotazy, zastává funkci WebSocket [21] serveru a má transformační funkce vhodné pro DMD dispečink. Z pohledu DMD dispečinku lze buďto poslat HTTP dotaz přímo na API [22] rozhraní, či se připojit přes WebSocket a dále pak zavolat požadovanou funkci s parametry. API rozhraní dotaz zpracuje, získá z databáze požadované dokumenty, transformuje je do požadované podoby a odešle zpět. Tímto se docílí toho, že v dispečinku jsou aktuální informace, které lze jakkoli upravovat, avšak v databázi IES serveru jsou stále uložena původní surová data. Nakonec je vhodné uvést shrnutí přínosů celého IES serveru:

- Veškeré změny firemních katalogů a jiných informací se okamžitě dostanou do všech mobilních zařízení.
- Nově vytvořené a upravené plány se okamžitě dostanou do zařízení s plánem spojeným.
- Jakákoliv práce s plánem v IES, tedy přijetí, splnění, přeskočení plánovaného režimu či další se okamžitě projeví v dispečinku.

- Průběh stazky v IES lze také synchronizovat. V dispečinku lze sledovat aktuální stav daného vozidla, jeho polohu, stav plnění plánu či další.
- Nově vytvořený bod zájmu se nejen dostane do dispečinku, ale zároveň i do ostatních mobilních zařízení ihned po založení.
- Lze přes tento datový tok zprovoznit určitou formu komunikace mezi dispečerem a řidičem.
- Data jsou uložena a zálohována ve své původní podobě na IES serveru.
- Data i IES serveru jsou převáděna do požadované podoby pro dispečink beze změny původních dat.
- Řešení umožňuje napojení více instancí dispečinku na jeden IES server.



Obrázek 7 - Schéma synchronizace dat v reálném čase mezi aplikací IES a dispečinkem

2.8.2.3. Multimodální interakce

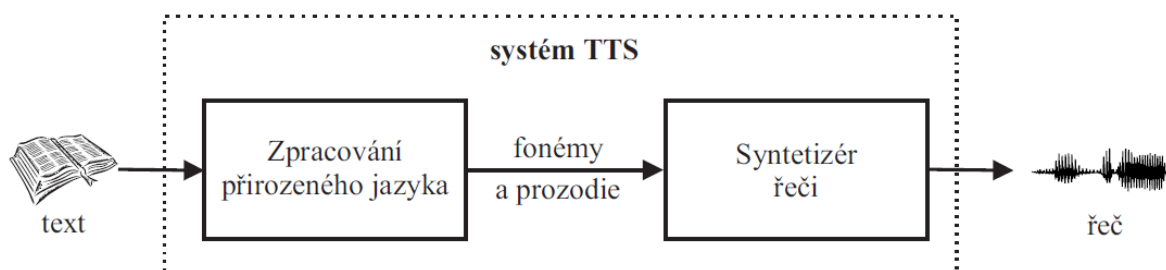
Pod pojmem multimodální interakce se obecně rozumí ovládání nějakého zařízení za použití více způsobů interakce zároveň. Příkladem může být právě ovládání mobilního zařízení. Prvním způsobem je ovládání stiskem tlačítka či dotykem na dotykovou obrazovku. Dalším způsobem může být ovládání hlasem, kdy zařízení reaguje na hlasové pokyny. Popřípadě by mohlo být zařízení ovládáno pohybem očí či pohybem ústy. Tato možnost ovládání více způsoby by přinesla mnoho výhod, jelikož pro každého jedince je pohodlnější,

výhodnější, či lépe srozumitelný jiný způsob interakce. Uživatel si tak může vybrat, jak bude zařízení ovládat a to vzhledem k aktuálním okolnostem, kde v některých situacích je výhodnější využít jiný způsob ovládání. Navíc je možné, že uživatel některý způsob interakce nemůže využívat a i přesto je mu umožněno zařízení používat. Multimodální ovládání je mnohem pružnější a přizpůsobivější větší škále uživatelů a prostředí.

3. Řečové technologie

3.1. Syntéza řeči z textu

Syntéza řeči, neboli převod textu na řeč, je celosvětově známa pod zkratkou TTS [23] - Text To Speech. Před popisem implementace je vhodné uvést způsob, jakým TTS funguje. Základem syntézy řeči z textu je syntetizér řeči, který podle vhodné vstupní informace uměle vytváří řeč. Vhodná vstupní informace se dá rozdělit na fonetickou a prozodickou. Fonetická informace je posloupnost fonémů, která reprezentuje vstupní text ve formě hlásek [24, 38]. Prozodická informace popisuje průběh řeči, nebo-li její rychlost, melodii, intenzitu a další. Díky těmto dvěma informacím je syntetizér schopen tvořit otázky, konce vět a další tvarování řeči tak, jak je v lidské řeči přirozené. K získáním těchto dvou druhů informací je nutné předzpracování přirozeného jazyka, které vstupní text analyzuje a odhadne fonetické i prozodické informace. Syntéza řeči z textu je obsáhlý problém, jelikož je potřebné, aby generovaná řeč zněla pro posluchače přirozeně a nestrojově, jako by ji vyřkla jiná osoba. Dále bude uveden způsob fungování korpusově orientované syntézy řeči, jelikož je tento způsob využit v IES.



Obrázek 8 - Schéma systému TTS. Převzato z [24].

3.1.1. Korpusově orientovaná syntéza řeči

V tomto přístupu je hlavně využit obsáhlý řečový korpus, který představuje základní řečový materiál, ze kterého lze řeč tvořit. Často je rozdělen do jednotlivých fonémů, které jsou řádně anotovány. Tento přístup je dodnes nejvíce využíván, jelikož přináší velmi dobré výsledky. Kvalita syntetizované řeči velmi závisí na kvalitě samotného korpusu. Jednotlivé fonémy by měly být nahrány v co nejčistší formě bez šumu a okolních ruchů. Také je vhodné aby byl korpus nahrán od jednoho mluvčího, pro co nepřirozenější výsledný tvar řeči [24].

Tento přístup využívá většinou tři základní metody syntézy řeči. První metodou je syntéza s jednou instancí řečových jednotek, která využívá malý korpus jednotlivých řečových segmentů. Při syntéze se v tomto případě vždy vybere pouze jeden reprezentant daného fonému či řečové jednotky. Při syntéze se vzhledem ke vstupní fonetické a prozodické informaci vybere z řečového korpusu posloupnost odpovídajících reprezentantů. V dalším kroku se musí tato posloupnost modifikovat tak, aby odpovídala požadované prozodické vstupní informaci. Také je nutné využít určité vyhlazování při spojování jednotlivých reprezentantů.

Další využívanou metodou je syntéza výběrem jednotek. Ta naopak využívá korpus, obsahující několik reprezentantů pro jednotlivé řečové jednotky. Tento přístup pak využívá metody pro výběr nejvhodnějšího reprezentanta dané jednotky vzhledem ke vstupním informacím. Tyto metody musí být rychlé, aby byla zajištěna syntéza v reálném čase, a zároveň musí zajistit výběr takového reprezentanta ze všech možných, aby výsledná řeč zněla co nejpřirozeněji. V neposlední řadě je nutné u tohoto přístupu zajistit, aby v případě, kdy nebyl nalezen vhodný reprezentant, byl ten nejvhodnější modifikován tak, aby vyhovoval požadavkům.

Posledním metodou korpusově orientované syntézy řeči je parametrická syntéza. Tato metoda je od předchozích nejvíce odlišná, jelikož nepracuje přímo s reprezentanty řečových jednotek, ale s průměrnými statistickými modely těchto jednotek. U této metody se využívají hlavně skryté Markovovy modely (zk. HMM [25]). V HMM se jako řečové jednotky používají fonémy a řeč je tak popsána jako posloupnosti akustických realizací fonémů. Syntéza pak spočívá v modelování jednotlivých fonémů pomocí třístavových HMM.

3.2. Rozpoznávání řeči

K rozpoznávání řeči, nebo-li k převodu mluveného slova na odpovídající text, se využívá systém automatického rozpoznávání řeči známý pod zkratkou ASR [26] - Automatic Speech Recognition. Myšlenka rozpoznávání lidské řeči se začala rozvíjet v druhé polovině 20. století, kdy započal významný rozvoj elektronických zařízení. Převod řeči na odpovídající text přináší mnoho využití v mnoha sférách běžného života. Hlavními příklady je psaní textu

hlasem, ovládání různých zařízení hlasem, ověření totožnosti hlasem, či usnadnění života sluchově handicapovaným lidem.

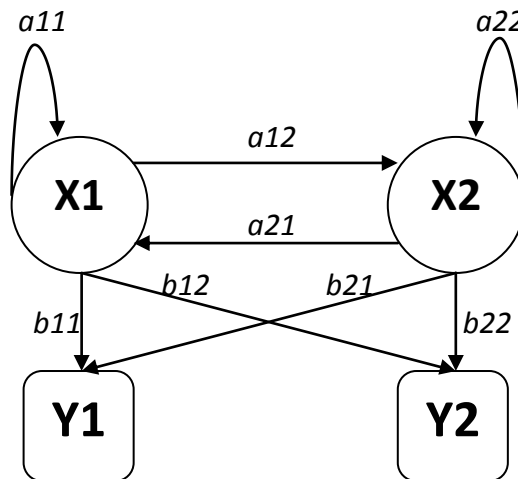
Přesto, že systémy automatického rozpoznávání řeči jsou ve vývoji již delší dobu, do dnes nejsou tyto systémy zcela bezchybné. Hlavním důvodem je fakt, že každý člověk vytváří řeč různým způsobem, různou rychlostí a intonací hlasu. Již základním problémem je konkrétní fyzický stav vnitřních orgánů mluvícího, jehož promluvu chceme rozpoznat. Hlavními orgány tvořícími řeč jsou plíce, dýchací svalstvo, hlasivky, nosní a ústní dutina a ústa. Ačkoli se zdá, že všichni lidé mají tyto orgány stejné, u každého člověka se tyto orgány s určitou odchylkou liší a jsou tak svým způsobem unikátní. Již z této podstaty každý člověk vytváří určitým způsobem unikátní řeč. Zásadní vlastností systému ASR musí proto být univerzálnost pro různé lidi. Dalším zásadním problémem v rozpoznávání řeči je to, že řečník mluví různými lingvistickými jazyky, různou intonací hlasu, různou rychlostí, řeč ovlivňují emoce, vady řeči a v neposlední řadě i minulost člověka, jelikož od dětství se učíme mluvit hlavně od okolí, ve kterém žijeme a ve kterém se používá určitý „slang“.

V dnešní době existuje několik metod zpracování a rozpoznávání řeči, které systémy ASR využívají. Hlavními dvěma směry je přímá analýza řečových signálů a statistické rozpoznávání řeči [39]. Přímá analýza řečových signálů využívá kódování tvarů vln, Fourierovu transformaci, lineární prediktivní analýzu, homomorfní zpracování řeči, fonetickou analýzu nebo vektorovou kvantizaci. Statistické rozpoznávání řeči využívají skryté Markovovy modely. Tyto metody používají akustický procesor, který pomocí transformace signálu převádí řeč na posloupnost příznaků a lingvistický dekodér, který tyto posloupnosti transformuje na jednotlivá slovní spojení. Vzhledem k využití těchto statických metod rozpoznávání řeči v IES je vhodné uvést způsob jejich fungování.

3.2.1. Statistické rozpoznávání řeči a skryté Markovovy modely

Statistické modely [39] využívající skryté Markovovy modely (HMM) definují množinu stavů určitého objektu a pravděpodobnosti vzájemných přechodů mezi těmito stavy. Pokud je HMM model předem natrénován, jsou pravděpodobnosti těchto přechodů známy a je možné v reálném čase odhadovat následující stav, do kterého objekt přejde. Pokud však natrénován není, jsou tyto pravděpodobnosti neznámé a tedy skryté. To je využívané

v rozpoznávání řeči, kdy k predikci následujícího slova stačí znát pouze současné rozpoznané slovo [27, 38]. U těchto metod je výhodné, že průběh řeči je možné popsat pravděpodobnostním modelem a HMM model lze pak chápat jako konečný automat, jehož příklad je zobrazen na obrázku 9. Zde je dvoustavový konečný automat popsán stavy x , možnými výsledky y a pravděpodobnostmi přechodů a a b .



Obrázek 9 - Příklad dvoustavového HMM modelu.

4. Multimodální ovládání Inteligentní Elektronické Stazky

Možnost ovládat IES více způsoby je velmi důležitá z hlediska charakteru využití. Vzhledem k tomu, že řidič vozidla využívající IES je s vozidlem v pohybu, je z právního a bezpečnostního hlediska nepřijatelné, aby ručně ovládal zařízení, na kterém je IES spuštěna. Řidič by tak v případě, kdy lze ovládat IES pouze ručně, musel vždy zastavit vozidlo, což je v praxi nereálné. Bylo proto nutné vytvořit více možností jejího ovládání.

Základní formou ovládání, jak již bylo řečeno, je ruční ovládání dotykem na obrazovku mobilního zařízení. Tato forma je ideální pro vyplňování údajů, rychlou práci s IES a interakci, pokud není vozidlo v pohybu. Vzhledem k automatizaci a inteligenci IES se dá říci, že druhou formou ovládání je nečinnost. Příkladem může být, že pokud v určitém místě ukončujeme nějaký režim vozidla, máme možnost buďto režim ukončit ručně stisknutím konkrétního tlačítka, či můžeme tento krok přeskocit a začít odjíždět. Automatizační jádro pak rozpozná odjezd z bodu zájmu, přechod do režimu jízda a předchozí režim samo ukončí s přepočítaným časem konce. Nečinnost však není ideální forma ovládání, má největší pravděpodobnost chybného výsledku.

Proto vznikl požadavek na třetí formu ovládání a to takovou, aby řidič mohl IES ovládat i během řízení vozidla. Takovou formou je ovládání hlasem. Pokud by IES reagovala na hlasové povely v součinnosti s ručním ovládáním, řidič by vždy měl možnost zvolit si, jaký způsob využije. Ovládání by tak bylo jednodušší a srozumitelnější. Pokud bychom k této formě ovládání navíc přidali schopnost převádět text na řeč, mohl by řidič s IES vést jednoduchý dialog ve formě otázka-odpověď. Možnost nejen příkazů, ale i možnost vést dialog mezi řidičem a IES byla proto hlavním cílem při vývoji možnosti ovládání hlasem.

4.1. Hlasové příkazy a dialogy v Inteligentní Elektronické Stazce

Před samotným vývojem hlasového ovládání bylo nutné provést analýzu a přesně určit, jaké možnosti má hlasové ovládání přinést a jak má pracovat. Byly vytvořeny vhodné slovníky pro určité situace. Je velmi vhodné učit jaké příkazy, či odpovědi mohou být od řidiče v různých situacích. Například při výběru typu režimu je velmi nepravděpodobné, že by řidič řekl například slovo „Ano“, jelikož je tato situace nejen nelogická, ale neexistuje na ni

ani ideální reakce. Některé slova proto v určitých situacích nemusíme rozpoznávat a složitost vývoje tak rozdělíme na menší, jednodušší části.

4.1.1. Globální příkazy pro navigaci a spuštění akce

Na obrázcích 3 a 4 v lze vidět, že základní grafické zobrazení IES je rozvrženo do několika záložek. Každá záložka zobrazuje jiné informace a nabízí různé možnosti volby. Pro řidiče je vhodné mít možnost, navigovat se v tomto zobrazení pomocí hlasu a také spouštět kdykoli některé akce v rámci celé aplikace. Po důkladném zvážení byly sepsány globální příkazy, které by měly být přístupny vždy, v rámci celé aplikace:

Navigace v hlavním okně - Příkazy pro přepnutí na určitou záložku v hlavním okně aplikace.

- *Souhrn*
- *Zakázky*
- *Stroj*
- *Stazka*
- *Menu*
- *Mapa*

Navigace v listu - V IES jsou mnohdy zobrazovány informace ve formě vertikálně posunovacího listu a je vhodné umožnit posunutí listu pomocí hlasu.

- *Předchozí*
- *Další*
- *Nahoru*
- *Dolu*

Příkazy dle aktuální záložky - V každé záložce jsou jiné volby relativní k záložce. Nicméně je vhodné některé tyto volby umožnit zvolit hlasem, i když není záložka aktivní.

- Záložka souhrn
 - *Změna režimu* - Začátek, konec, či změna aktuálního režimu vozidla.
 - *Výjimka* - Vyvolání dialogu možných výjimek. Například porucha vozidla.
- Záložka zakázky

- *Aktualizovat plán* - Online aktualizace zakázek a plánu pro nynější vozidlo a řidiče.
- Záložka stroj
 - *Odsouhlasit stav* - Potvrzení správných údajů o vozidle. Například, zda-li je vypočtený stav kilometrů správný. Tato volba je možná pouze, pokud je záložka stroj aktivní.
- Záložka stazka
 - *Odsouhlasit stazku* - Odsouhlasení vytvořené stazky a správnosti údajů.
- Záložka menu
 - *Volání* - Rychlé přepnutí do seznamu kontaktů pro telefonní hovor.
 - *Přepnout aplikaci* - Přepnutí na předem definovanou jinou aplikaci, například e-mailového klienta.
 - *Poznámka* - Vytvoření dialogu pro zápis poznámky k aktuálnímu režimu.
 - *Výměna řidičů* - Možnost zvolit řidiče, který se střídá s aktuálním v rámci jedné stazky.
 - *Založit zakázku* - Ruční založení zcela nové zakázky, například v situaci, kdy dispečer dopředu nezná objednavatele zakázky.
 - *Soukromá jízda* - Přepnutí do režimu soukromá jízda. Tento režim je výjimečný, a proto není v hlavním seznamu režimů vozidla.
 - *Uspání IES* - Dočasné přepnutí IES do spánkového režimu, například při obchodní schůzce. Tato volba je přístupná pouze vybraným řidičům.
 - *Konec Stazky* - Přepnutí do režimu ukončování stazky, kdy řidič musí ještě odsouhlasit koncové informace o vozidle a doplnit potřebné informace do vytvořené stazky.
 - *Stazka nekončí* - Možnost vrátit se zpět do vytváření stazky při chybné volbě Konec Stazky.

4.1.2. Odpovědi na otázky a příkazy v dialozích

Pro maximální přehlednost a srozumitelnost je aplikace stavěna tak, že hlavní okno aplikace je vždy viditelné a pro zobrazení, určení či doplnění informací jsou využívána dialogová okna. Vzhledem k již zmíněné výhodnosti rozdělení ovládání hlasem na určité

menší části je využití dialogových oken ideální. Při jeho zobrazení víme, jaké informace jsou uživateli zobrazeny, jaké informace aplikace potřebuje, na co se má aplikace ptát a jak může řidič odpovědět.

Odpovědi na jednoduché dialogy - Během stazky se řidič setká s mnoha jednoduchými dialogy, které buďto pouze informují, či požadují odpověď *Ano/Ne*. V takové situaci IES zobrazí dialog, převede zobrazený text na řeč a vysloví jej. Následně začne naslouchat na odpověď řidiče. Tuto odpověď pak porovnává s redukováným slovníkem, konkretizovaným pro daný dialog. Ukázka těchto jednoduchých dialogů je na obrázku 10.

- *Ano*
- *Ne*
- *Zpět*
- *Ok*



Obrázek 10 - Ukázka informačního dialogu a dialogu s odpovědí *Ano/Ne*

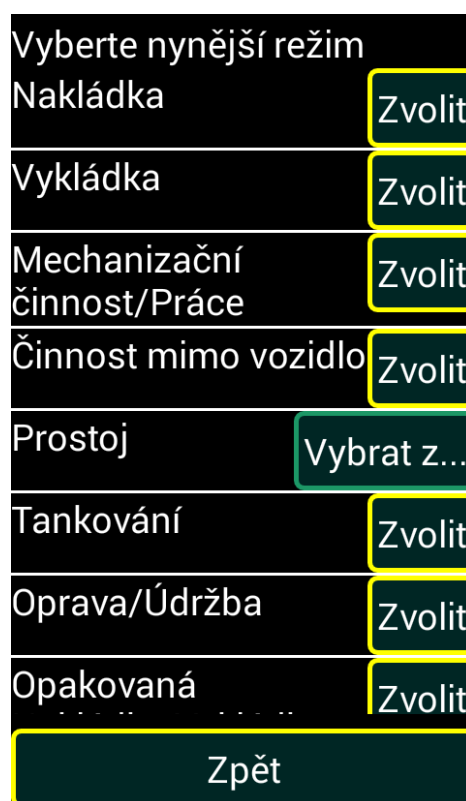
Odpovědi na dialogy s různými odpověďmi - Další skupinou jsou dialogy, které buďto mají na výběr více než dvě odpovědi, či jsou odpovědi složitější než pouze *Ano/Ne*. U těchto dialogů je opět převeden na řeč stejný text, jako je uživateli zobrazen.

- *Zkontrolovat*
- *Nestahovat*
- *Zkusit znovu*
- *Začít stazku*
- *Uložit a zavřít*
- *Pokračuje režim jízda*
- *Zkusit znovu*
- *Neodesílat a ukončit*

- *Naloženo*
- *Vyloženo*
- *Přidat*

Dialog výběru typu režimu - Tento dialog je jedním z nejčastějších, jelikož díky němu lze změnit aktuální režim vozidla, určit typ režimu již provedeného, avšak dosud neurčeného a další. Zde je již na řeč převedena konkretizovaná otázka, například „Vyberte nyní režim vozidla“. Mezi odpovědi patří všechny možné typy režimů včetně jejich poddruhů.

- Příprava před jízdou
- Jízda
- Nakládka
- Vykládka
- Mechanizační činnost/práce
- Činnost mimo vozidlo
- Prostoj
 - Bezpečnostní přestávka
 - Prostoj dopravce
 - Prostoj přepravce
 - Spaní v autě
 - Spaní mimo vozidlo
- Tankování
- Oprava/Údržba
- Opakovaná nakládka-vykládka
- Odstavení vozidla



Obrázek 11 - Dialog výběru typu režimu

4.1.3. Složitější příkazy zadávání a výběru

Při vyplňování informací se mnohdy dostaneme do situací, na které nelze vytvořit jednoduchý slovník odpovědí, jelikož je odpovědí nespočetně mnoho. Příkladem může být výběr vozidla dle SPZ ze všech firemních vozidel, vyplňování názvu nově založeného názvu či diktování poznámky. Při výběru položky ze seznamu sice dopředu známe množinu všech položek, avšak ty se mohou časem měnit, je proto nutné zajistit včasnou aktualizaci slovníku a dalších souvislostí. Složitější je také číselný slovník pro zadávání například stavu počítadla

ujetých kilometrů vozidla. Naopak při zadávání nového názvu místa či poznámky již vůbec nelze předpokládat odpověď. Je nutné využít rozpoznávání řeči s kompletním slovníkem daného jazyka, označován anglickou zkratkou LVCSR [28] (large-vocabulary continuous speech recognition). Následuje výpis příkladů složitějších příkazů:

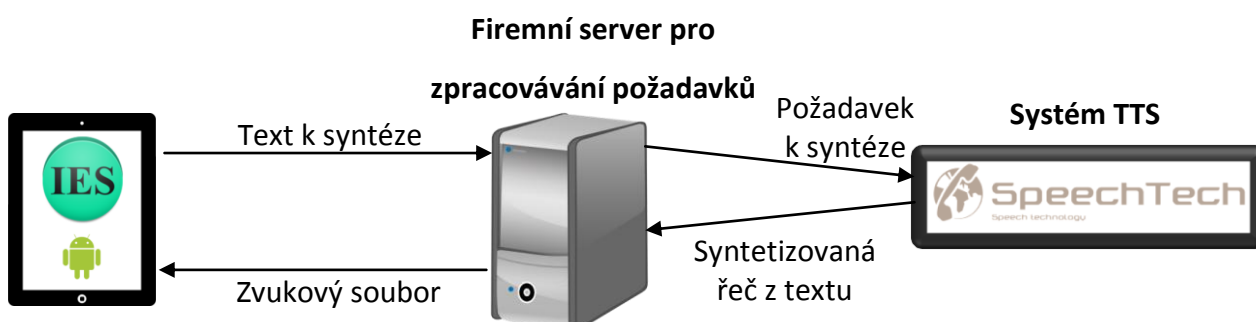
- Výběr z katalogu posádky
 - Příklad odpovědí: Jan Novák, Karel Dvořák.
- Výběr z katalogu tažných a přípojných vozidel
 - Příklad odpovědí: 4P5 0425, A24 1423.
- Výběr z katalogu bodů zájmu
 - Příklad odpovědí: Litice areál firmy, Plzeň slovanská 32, Dobřany Bögl a Krysl.
- Výběr z katalogu substrátů
 - Příklad odpovědí: Štěrk 31/25, Cihly na paletách, Dřevo souška.
- Výběr z katalogu zboží k nákupu
 - Příklad odpovědí: Natural 95, Diesel Premium, Chladicí kapalina.
- Výběr z katalogu čerpacích stanic
 - Příklad odpovědí: OMW, Agip, Benzina.
- Zadávání stavu ukazatelů, množství a cena - stav kilometrovníků, stav paliva v nádrži, počet motohodin, množství substrátu, množství nakoupeného zboží, cena a další.
 - Příklad odpovědí: 230 524, 2, 24.50, 158.
- Zadávání názvu nového bodu zájmu či poznámky - jakýkoliv text
 - Příklad odpovědí: České Budějovice poblíž nádraží, V kamenolomu měli zavřeno.

4.2. Implementace syntézy řeči z textu do IES

Při implementaci systému TTS do IES byla velmi využita spolupráce se Západočeskou univerzitou v Plzni a její „spin-off“ firmou SpeechTech s.r.o. [29], která v rámci projektu poskytla významnou podporu. Pro syntézu řeči z textu na mobilních zařízeních se tak naskytly hned dvě možnosti, jak tento problém řešit.

4.2.1. Online syntéza řeči na mobilním zařízení

První možností byla možnost online napojení na systém TTS běžící na serverech firmy SpeechTech. Pro testovací účely byl vytvořen server pro zpracování požadavků na převod textu do řeči. Schéma fungování popisuje obrázek 12. Mobilní zařízení, pokud potřebuje syntézu, pošle čistý text na firemní server. Server přepoše požadavek na syntézu systému TTS běžícím u firmy SpeechTech. Jakmile je text převeden, systém TTS vrací řeč ve formě zvukového souboru. Firemní server pak daný soubor přepoše zpět mobilnímu zařízení, které ho přehraje jako hudbu na pozadí. Rychlost syntézy je závislá nejen na rychlosti přenosu dat přes internet, ale i na rychlosti procesu syntézy v systému TTS. Ukázka rychlosti syntézy řeči z textu v systému TTS je ukázána v tabulce 1, rychlost přenosu dat je velmi proměnná.



Obrázek 12 - Schéma online syntézy řeči na mobilním zařízení.

| Text použitý k syntéze | Rychlost syntézy[s] |
|--|---------------------|
| <i>Zadejte název.</i> | 0.8 |
| <i>Do konce bezpečnostní přestávky zbývá 10 minut.</i> | 1.2 |
| <i>Nacházíte se v místě: Dobřany - Lom Dubová Hora, kde je plánovaná nakládka v zakázce č. 87629. Začíná nyní tento režim?</i> | 2.6 |

Tabulka 1 - Rychlost syntézy řeči z textu v systému SpeechTech TTS

4.2.2. Offline syntéza řeči na mobilním zařízení

Na operačním systému Android je již od základu instalován modul převodu textu na řeč od firmy Google Inc. [30]. Tento modul podporuje několik hlavních celosvětových jazyků a je možné ho zcela zdarma využívat. Bohužel firma Google zatím češtinu do tohoto modulu neimplementovala a v nejbližší době se nechystá. Firma SpeechTech se však vývojem

podobného modulu podporujícího český jazyk zabývá, což je pro tento projekt velmi výhodné. Další možností proto bylo využít specializovanou aplikaci SpeechTech TTS pro operační systém Android. Tato aplikace je sice stále v testovací fázi, ale v rámci projektu bylo umožněno ji využívat. Aplikace využívá korpusový přístup k syntéze řeči a částečně nahrazuje systémové jádro pro syntézu řeči z textu operačního systému Android. Po instalaci aplikace je možné stáhnout několik různých korpusů, rozdělených dle osob, které je vytvářely. Po stáhnutí nějakého korpusu je již systémově možné využívat offline syntézu české řeči. Vzhledem k tomu, že tento modul je systémový, tedy je využíván v rámci celého operačního systému, je možné jej využívat z jakékoli aplikace běžící na daném operačním systému. V IES byla vytvořena třída obsluhující právě tento modul pro převod textu na řeč a tím byla IES nepřímo napojena na offline systém TTS firmy SpeechTech. V aplikaci pak po správném nastavení stačí zavolat konkrétní funkci a jako parametr jí předat text pro převod. Operační systém pak automaticky přehraje výslednou řeč na pozadí.

4.3. Rozpoznávání řeči v IES

V IES je třeba vyřešit dva různé úkoly v rámci rozpoznávání řeči. Prvním, jednodušším úkolem je rozpoznávání příkazů z konkrétního, ne příliš velkého slovníku možných příkazů. Vzhledem k tomu, že je tento slovník k dispozici, hlavním úkolem systému ASR je určit pravděpodobnosti vyslovení daných příkazů. Pokud byl opravdu vysloven nějaký příkaz ze slovníku, bude jeho pravděpodobnost maximální a IES tak může na tento příkaz zareagovat. U příkazů mimo slovník či nejistotě, zda-li byl rozpoznán správný příkaz, by IES nereagovala či by požadovala zopakování příkazu. Mnohem složitějším problémem je však druhý úkol, kdy uživatel IES potřebuje vybrat položku ze seznamu či zadat spojitý, předem neznámý text, například při objasnění důvodu odchýlení se z plánu jízdy. Vzhledem k tomu, že v této situaci nemáme dopředu žádnou informaci o řečeném textu, je tato úloha mnohem náchylnější k chybnému rozpoznání textu, či nepochopení jeho kontextu. Při implementaci systému ASR do IES byl nejprve řešen výběr příkazu ze slovníku a poté možnost zadání čistého textu.

4.3.1. Google ASR

Společnost Google Inc. má významné postavení i na poli automatického rozpoznávání řeči. Jejich systémy ASR lze využívat nejen v operačním systému Android, ale i na webových prohlížečích na běžných počítačích. Společnost Google Inc. určitým způsobem umožňuje používání jejich systému ASR a proto byla provedena analýza a možnost jejich využití.

V operačním systému Android verze API 8.0 [31] a vyšší je implementována možnost využití systémového modulu rozpoznávání řeči. Tento modul využívá odesílání datového toku na systém ASR a je proto nutné připojení k internetu. V operačním systému Android je možnost stáhnout si obsáhlý balík potřebných dat k offline rozpoznávání řeči, avšak dosud neexistuje balík pro český jazyk a lze tak využít pouze online rozpoznávání. Do IES bylo implementováno využití tohoto modulu a byl dále otestován. Rozpoznávání s využitím tohoto modulu funguje vcelku dobře, avšak hlavním problémem je fakt, že jeho používání není tak otevřené, jak bychom potřebovali. Veškeré použití modulu závisí na následujícím zdrojovém kódu:

```
/**
 * Method for start speech recognition
 **/
private void startSpeechRecognition() {
    Intent intent = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL, "cz-CS");
    startActivityForResult(intent, RESULT_SPEECH);
}
/**
 * Method for receive result of speech recognition
 * */
protected void onActivityResult(int requestCode, int resultCode, Intent
data) {
    if (requestCode == REQ_CODE_SPEECH_INPUT && resultCode == RESULT_OK
&& data != null) {
        ArrayList<String> result =
data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
        Log.i("ASR result", result.get(0));
    }
}
```

V tomto zdrojovém kódu lze vidět, že modul se chová jako samostatná jednotka a jeho možnost nastavení či úpravy je minimální. V metodě *startSpeechRecognition* se pouze určí rozpoznávaný jazyk a zavolá se spuštění modulu. V metodě *onActivityResult* se následně testuje, zda-li je výsledek od tohoto modulu, a poté se vypíše do konzole. Ačkoliv je tento modul volně k použití, jeho způsob fungování a pokročilé možnosti nastavení jsou pro aplikace třetích stran uzavřeny a tím si společnost Google Inc. tímto způsobem svůj systém ASR chrání. Navíc bylo zjištěno, že během rozpoznávání tento modul aplikaci IES překrývá. Ačkoliv je tento modul funkční a výsledky jsou v převážné většině správné, je využití tohoto modulu nežádoucí právě z důvodu nemožnosti jakékoli úpravy či pokročilého nastavení.

Další možností je využití Google Speech Api, což je interface pro připojení přímo na systémy ASR společnosti Google. Tímto způsobem se dá obejít využití modulu a spravovat celý proces rozpoznávání řeči přímo z aplikace IES. V IES se uloží nahrávaná řeč z mikrofону do souboru ve formátu *.flac* [32], tento soubor se odešle přímo na systém ASR a systém pak vrací zpět možné výsledky s informací o jejich důvěryhodnosti. Celý tento proces se odehrává na pozadí a lze již využít pokročilé možnosti úpravy, například odfiltrování šumu v nahrávce ještě před odesláním na systém ASR. Bohužel v tomto procesu stále není možnost volby vlastního omezeného slovníku slov, a proto je chybovost výsledku vždy větší, než by mohla být s omezeným slovníkem. Využívání Google Speech Api je zdarma pouze do 50 požadavků na zpracování za jeden den. Pokud bychom potřebovali zpracovávat více požadavků, je nutné si tyto služby pronajímat.

4.3.2. Systém ASR Západočeské univerzity v Plzni

Vzhledem ke spolupráci na tomto projektu se Západočeskou univerzitou v Plzni bylo využito možnosti využití systému ASR, který Západočeská univerzita vyvíjí. Tento systém pracuje na statistickém přístupu rozpoznávání řeči se skrytými Markovovi modely. Možnosti nastavení a upravení na míru jsou v tomto případě velké a proto je volba tohoto ASR přínosnější oproti využití systému ASR od firmy Google Inc. Průběh procesu rozpoznávání je podobný, na mobilním zařízení se nahraje zvuk z mikrofónu do dočasného souboru. Tento soubor se pošle na server se systémem ASR, spolu s ním se však pošle požadovaný slovník, ze kterého se má rozpoznávat a další určující nastavení pro maximalizaci správného rozpoznání. Server pak zpět vrací rozpoznávaný text ze slovníku a jeho věrohodnost.

4.3.3. Implementace ZČU ASR do IES

Před samotnou implementací využití systému ASR Západočeské univerzity bylo nutné vytvořit nastavení pro rozpoznávač, které je vhodné pro tuto úlohu. Nejprve byl vytvořen slovník možných příkazů, již zmíněný v kapitole 4.1. Pro tento slovník byly vytvořeny odpovídající gramatiky. Dále byly vytvořeny sady nahrávek od tří osob a to nejprve v naprostém tichu, aby byl nahraný zvuk bez jakéhokoliv okolního rušení. Dále byly nahrány sady během jízdy v nákladním či osobním vozidle. U první osoby byla také vytvořena sada nahrávek se simulovaným ruchem okolí, vloženým do nahrávek bez ruchu. Tím byla vytvořena data potřebná k porovnání rozpoznávání v ideální a reálné situaci. Data byla následně analyzována, rozdělena do potřebné formy a byly k nim vytvořeny potřebné anotace, tím byly vytvořeny gramatiky ve formátu ESGF [33], nebo-li bezkontextová gramatika reprezentována pravidly ESGF. Následně je uveden tvar ESGF gramatiky pro skupinu obecných příkazů:

```
#ESGF V1.0;
grammar IesGramObecne;

public <IesGramObecne> = <BEGIN_GRAMMAR> <VIRTUAL> ( <VIRTUAL>
    <IesPrikazyObecne> <VIRTUAL> )+ <END_GRAMMAR>;

<IesPrikazyObecne> = (
    Souhrn |
    Zakázky |
    Stroj |
    Stazka |
    Menu |
    Mapa |
    Nahoru |
    Dolu |
    (Odsouhlasit stav) |
    (Aktualizovat plán) |
    (Změna režimu) |
    (Odsouhlasit stazku)
);
```

Po vytvoření potřebných gramatik začalo testování rozpoznávání řeči na všech nahraných sadách. Vzhledem k velké možnosti nastavení rozpoznávače bylo nutné najít

vhodné nastavení pro správné rozpoznávání řeči v situaci, kdy je na pozadí slyšet zvuk motoru nákladního vozidla, mechanizačního stroje či dalších zvuků, které se běžně vyskytují během dopravních situací. Výsledky z rozpoznávání jsou zobrazeny v tabulce 2 a 3. V tabulkách jsou zobrazeny jednotlivé sady od řečníků a k nim výsledné údaje o správnosti a přesnosti rozpoznávání, uvedené v procentech. Výsledky jsou zobrazeny pro všechny vytvořené gramatiky, konkrétně pro kompletní gramatiku s rozdělenými slovy, kompletní gramatiku se spojenými slovy, gramatiku pro dialogy, gramatiku pro ovládání menu, gramatiku obecných příkazů a gramatiku při výběru typu režimu vozidla či stroje. U kompletní gramatiky s rozdělenými slovy byly použity všechny příkazy ze slovníku a dané příkazy byly rozděleny dle jednotlivých slov. Například slovní spojení „Odsouhlasit stazku“ bylo rozděleno na dvě jednotlivá slova, která mohla být rozpoznány samostatně. Jedná se tedy o ekvivalent 0-gramového jazykového modelu [34]. V kompletní gramatice se spojenými slovy byly také použity všechny příkazy, avšak v gramatice jsou obsaženy pouze celé příkazy, tedy i slovní spojení.

| Výsledky rozpoznávání řeči systémem ASR Západočeské Univerzity | | | | |
|---|------------------------------------|--------------------------------|----------------------------------|--------------------------------|
| Gramatika | Kompletní - rozdělená slova | | Kompletní - spojená slova | |
| | Správnost rozpoznání [%] | Přesnost rozpoznání [%] | Správnost rozpoznání [%] | Přesnost rozpoznání [%] |
| Sada (Hlas) | | | | |
| Sada 1 - Čistý zvuk | 84.38 | 83.13 | 96.39 | 95.18 |
| Sada 1 - Simulovaný ruch | 44.58 | 42.17 | 75.90 | 69.88 |
| Sada 1 - Reálný ruch | 68.67 | 67.47 | 90.36 | 89.16 |
| Sada 2 - Čistý zvuk | 89.16 | 89.16 | 98.80 | 98.80 |
| Sada 2 - Reálný ruch | 61.45 | 61.45 | 84.34 | 91.93 |
| Sada 3 - Čistý zvuk | 93.98 | 92.77 | 98.80 | 98.80 |
| Sada 3 - Reálný ruch | 32.53 | 32.53 | 73.49 | 69.88 |
| Průměr | 67.82 | 66.95 | 88.29 | 87.66 |

Tabulka 2 - Výsledky rozpoznávání pro kompletní gramatiky

| Výsledky rozpoznávání řeči systémem ASR Západočeské Univerzity | | | | | | | | |
|--|-----------------|--------------|---------------|--------------|----------------|--------------|----------------|--------------|
| Gramatika | Skupina Dialogy | | Skupina Menu | | Skupina Obecné | | Skupina Režimy | |
| | Správnost [%] | Přesnost [%] | Správnost [%] | Přesnost [%] | Správnost [%] | Přesnost [%] | Správnost [%] | Přesnost [%] |
| Sada 1 - Čistý zvuk | 90.91 | 90.91 | 100 | 100 | 100 | 100 | 100 | 100 |
| Sada 1 - Simul. ruch | 86.36 | 86.36 | 100 | 100 | 81.25 | 75 | 93.10 | 93.10 |
| Sada 1 - Reálný ruch | 86.36 | 86.36 | 100 | 100 | 93.75 | 93.75 | 100 | 100 |
| Sada 2 - Čistý zvuk | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| Sada 2 - Reálný ruch | 72.73 | 72.73 | 93.75 | 93.75 | 87.50 | 87.50 | 100 | 100 |
| Sada 3 - Čistý zvuk | 100 | 95.45 | 100 | 100 | 100 | 100 | 100 | 100 |
| Sada 3 - Reálný ruch | 50 | 50 | 100 | 87.50 | 81.25 | 81.25 | 100 | 100 |
| Průměr | 83.76 | 83.11 | 99.10 | 97.32 | 91.96 | 91.07 | 99.01 | 99.01 |

Tabulka 3 - Výsledky rozpoznávání pro gramatiky rozdělené do skupin

Z výsledků lze vidět, že kompletní gramatika se spojenými slovy je vždy přesněji rozpoznána, než-li gramatika se spojenými slovy. Vzhledem k menší množině možných výsledků u rozdělených gramatik lze vidět, že i při velkém ruchu na pozadí je rozpoznávání převážně úspěšné, a proto i tento směr použití správný. Pro vyhodnocení výsledků byl využit program Hresult. Pro účely porovnání byly nahrané sady odeslány pomocí Google Speech API [35] na systém ASR společnosti Google Inc. Tento systém ASR je určen pro rozpoznávání delších promluv a využívá gramatiku pro celý český jazyk (LVCSR). U tohoto ASR není možné uživatelské nastavení ani možnost použít vlastní gramatiku. Výsledky rozpoznávání jsou uvedeny v tabulce 4. Při analýze výsledků bylo zjištěno, že tento systém ASR, jelikož je určen

pro delší promluvy, vyhodnocuje krátké příkazy nepřesně. Pokud danému slovu neporozumí správně, buďto vrátí chybný výsledek, či žádný. U sad s okolním ruchem bylo časté, že u příkazů o jednom slově tento systém nevracel žádné výsledky. Proto jsou v této tabulce výsledky správnosti a přesnosti rozpoznávání rovné.

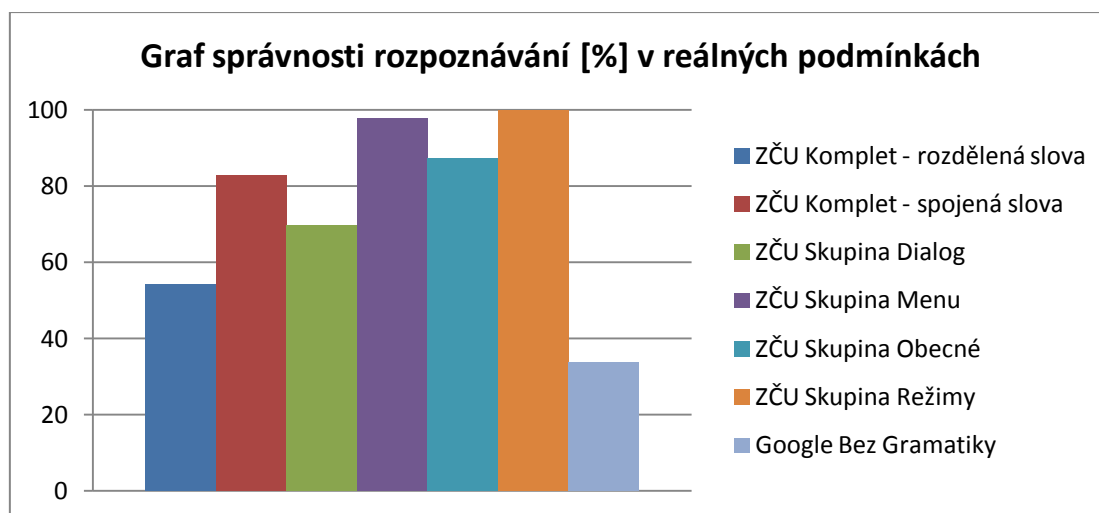
| Výsledky rozpoznávání řeči systémem Google Inc. | | |
|--|---|---------------------------------|
| Gramatika | LVCSR - Gramatika pro celý český jazyk | |
| | Správnost rozpoznávání [%] | Přesnost rozpoznávání[%] |
| Sada (Hlas) | | |
| Sada 1 - Čistý zvuk | 73.49 | 73.49 |
| Sada 1 - Reálný ruch | 33.73 | 33.73 |
| Sada 2 - Čistý zvuk | 73.49 | 73.49 |
| Sada 2 - Reálný ruch | 46.99 | 46.99 |
| Sada 3 - Čistý zvuk | 68.67 | 68.67 |
| Sada 3 - Reálný ruch | 20.48 | 20.48 |
| Průměr | 52.80 | 52.80 |

Tabulka 4 - Výsledky rozpoznávání systémem Google ASR

Následně byly použity pouze výsledky správnosti rozpoznávání u jednotlivých sad a přístupů a z nich vytvořené průměry. Tyto data lze vidět v tabulce 5. Z výsledků je jasné, že nejlepším přístupem je použití systému ASR Západočeské univerzity při použití gramatik rozdělených do skupin. U sad bez okolního ruchu byly výsledky téměř vždy zcela správné. U sad, kde byl na pozadí obsažen ruch, byly výsledky také velmi dobré, pouze u sady číslo 3 vyšel špatný výsledek u gramatiky využití v dialozích. Tato sada byla nahrávána z velké vzdálenosti od mikrofону, a proto není tento výsledek kritický. Výsledek rozpoznávání systému Google ASR je tak nízký, protože tento systém používá gramatiku pro celý český jazyk, což je nevhodné pro rozpoznávání krátkých příkazů. Průměry správnosti rozpoznávání jednotlivých přístupů při podmínkách v reálném provozu jsou pro lepší přehlednost zobrazeny v grafu 1.

| Správnost rozpoznávání [%] u jednotlivých přístupů a typů ASR | | | | | | | | |
|---|------------|------------|------------|-----------------|------------|------------|------------|-----------------|
| Sada(Hlas) Typ ASR | 1 bez šumu | 2 bez šumu | 3 bez šumu | Průměr bez šumu | 1 se šumem | 2 se šumem | 3 se šumem | Průměr se šumem |
| ZČU- Kompletní - rozdělená slova | 84.38 | 89.16 | 93.98 | 89.17% | 68.67 | 61.45 | 32.53 | 54.21% |
| ZČU - Kompletní - spojená slova | 96.39 | 98.80 | 98.80 | 97.99% | 90.36 | 84.34 | 73.49 | 82.73% |
| ZČU -Skupina Dialogy | 90.91 | 100 | 100 | 96.97% | 86.36 | 72.73 | 50 | 69.69% |
| ZČU -Skupina Menu | 100 | 100 | 100 | 100% | 100 | 93.75 | 100 | 97.91% |
| ZČU -Skupina Obecné | 100 | 100 | 100 | 100% | 93.75 | 87.50 | 81.25 | 87.5% |
| ZČU -Skupina Režimy | 100 | 100 | 100 | 100% | 100 | 100 | 100 | 100% |
| ASR-Google LVSR | 73.49 | 73.49 | 68.67 | 71.88% | 33.73 | 46.99 | 20.48 | 33.7% |

Tabulka 5 - Výsledné průměry správnosti rozpoznávání u jednotlivých přístupů a typů ASR



Graf 1 - Úspěšnost rozpoznávání v reálném provozu pro jednotlivé přístupy

4.3.4. Offline systém ASR

Hlavním problémem při využívání systému ASR je nutnost odesílání dat na server, kde tento systém běží. U mobilního zařízení je nutné být neustále připojen na internet, což je

nežádoucí nejen z důvodu placení mobilních dat, ale i z důvodu, že mobilní signál není vždy zcela stabilní. Pokud by se zařízení nacházelo na místě se slabým či žádným mobilním signálem, celý proces rozpoznávání řeči by tak byl buďto velmi pomalý či zcela nepřístupný. Proto je nutné mít možnost rozpoznávat řeč i v offline režimu, bez připojení na internet. Jak již bylo zmíněno, v operačním systému Android je možné využít offline režim, avšak dosud existuje pouze 15 balíčků pro několik hlavních celosvětových jazyků. Není jisté, kdy a zda-li bude vydán balík pro český jazyk a proto bylo opět přistoupeno ke spolupráci se Západočeskou univerzitou v Plzni. Pro rozpoznávání řeči offline je potřebný vysoký výpočetní výkon, který dosud není schopné mobilní zařízení zcela nabídnout. Proto je vhodným kompromisem možnost rozpoznávat řeč offline pouze z omezeného slovníku, to odpovídá požadavkům pro využití v IES. Převodění celého systému ASR, hlavně akustického procesoru a lingvistického dekodéru přímo do operačního systému Android je však velmi obsáhlý a časově náročný úkol, který je nad rámec této práce. V rámci projektu TA ČR číslo TA04031301 [4] se však s vývojem tohoto offline systému ASR počítá a bude nadále vyvíjen a později implementován do IES.

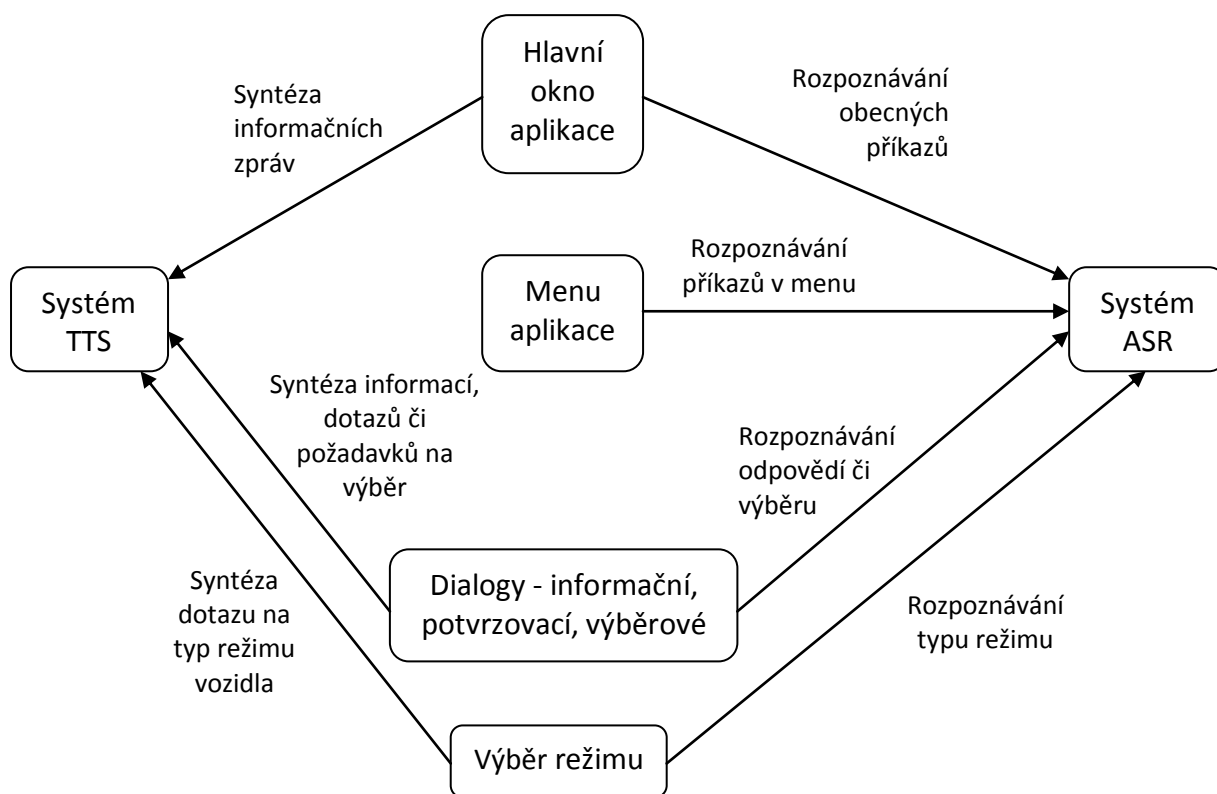
4.4. Shrnutí syntézy řeči z textu a automatického rozpoznávání řeči

Při implementaci systému TTS bylo možné volit z několika možností. První možností bylo začít vyvíjet vlastní systém TTS. To bylo v rámci projektu a možností firmy Hobl&Pech nežádoucí. Další možností bylo využít modul pro převod textu na řeč od firmy Google Inc., který je již v základu operačního systému Android. Tento modul však nepodporuje český jazyk a není jisté, kdy a zda-li vůbec začne. Vzhledem k tomu, že IES je směřována především na český trh, bylo nutné najít jiné řešení. Ideální bylo propojení se Západočeskou univerzitou v Plzni a firmou SpeechTech s.r.o., kde bylo možné využít online či offline syntézu řeči z textu. U online syntézy musí být mobilní zařízení neustále připojené na internet a celý proces syntézy, včetně odesílání a přijímání dat, trvá v řádu 1-15 vteřin. Naopak offline syntéza, zakomponovaná přímo do operačního systému, je zcela samostatná, má spolehlivý způsob použití a rychlost syntézy je okamžitá. Z těchto jasných důvodů byla zvolena cesta offline systému TTS firmy SpeechTech. Při produkci IES na trh se pak počítá s tím, že tento modul bude nabízen jako doplňková služba.

Při volbě systému automatického rozpoznávání řeči bylo také možné volit z několika možností. Opět bylo možné využít existující technologie od firmy Google Inc., které jsou zakomponovány v operačním systému Android. Při využití tohoto systému ASR by pro český jazyk muselo být zařízení neustále připojené k internetu, a vzhledem k výsledkům z testování by kvalitní rozpoznávání příkazů nebylo zaručeno. Z důvodu malé možnosti nastavení a přizpůsobení systému pro vlastní potřeby bylo od této možnosti opuštěno a místo toho byla, stejně jako v systému TTS, využita možnost spolupráce se Západočeskou univerzitou v Plzni. Tento systém nabízí maximální možnosti přizpůsobení, výsledky rozpoznávání jsou velmi dobré a do budoucna je počítáno s přesunutím zredukované verze tohoto systému ASR přímo na operační systém Android. Tím bude docíleno toho, že budeme schopni syntetizovat a rozpoznávat český jazyk na mobilním zařízení bez nutnosti připojení k internetu, což je vzhledem k povaze projektu velmi žádoucí.

4.5. Schéma využívání TTS a ASR v IES

Moduly pro převod textu na řeč a rozpoznání textu z řeči byly implementovány a připraveny na plné využití v IES. Pro maximalizaci správných výsledků jsou gramatiky pro rozpoznávání rozděleny do skupin, dle jejich využití v různých situacích. Konkrétně gramatika pro obecné příkazy, gramatika pro ovládání menu, gramatika pro informační a výběrové dialogy a gramatika pro výběr či určení režimu vozidla. Následně je zobrazené schéma využití těchto modulů při tvorbě elektronického záznamu v IES.



Obrázek 13 - Schéma propojení využívání systémů TTS a ASR v aplikaci IES.

Toto rozvržení do několika menších úkolů je výhodné nejen k minimalizaci možné chyby, ale i pro využití výpočetního výkonu daného zařízení. Dále je vhodné zmínit, že při reálných testech se velmi osvědčil i model, kdy je využíván pouze systém převodu textu na řeč. Příkladem může být informační dialog o nutnosti započítí bezpečnostní přestávky. Řidič totiž nemusí věnovat IES zrakovou pozornost, avšak když tato situace nastane, je nejen náležitě upozorněn, ale i informován bez nutnosti přestat sledovat vozovku.

5. Aktuální stav, specifikace a budoucnost IES

V době psaní této práce je většina aplikace IES plně funkční a náležitě otestována. Plánování v dispečinku, tvorba elektronického záznamu, odstranění nutnosti interakce obsluhy, napojení na sledovací jednotky vozidel, zpětná vazba k dispečerovi a zobrazení kompletního elektronického záznamu v dispečinku je zcela funkční a otestované reálnými jízdami v řádu stovek hodin. Multimodální ovládání IES je v testovací fázi, kdy je napojení na systémy TTS a ASR v provozu, ale je nutné před uvedením do distribuce tyto systémy náležitě otestovat a upravit do konečné fáze.

Aplikace IES je směřována na operační systém Android 2.2. - API 8.0 a vyšší. Díky tomu může IES běžet na 90% všech zařízení s tímto operačním systémem. Aplikace je psaná v programovacím jazyce JAVA. Obsahuje 133 tříd rozdělených do 16 balíčků. Má implementováno 204 statických metod a přibližně 474 privátních metod. Celá aplikace je pak napsána na 20 956 řádcích čistého kódu. Jednotlivé třídy a objekty v rámci aplikace jsou navzájem velmi provázané a závislé. Aplikace je pak také napojena na mnoho systémových prostředků operačního systému.

Do budoucna je počítáno s vývojem IES v řádu několika let, jelikož se při vývoji potvrzuje, že jsou stále směry, kde je možné IES možné udělat více automatizovanou a inteligentnější. Je počítáno s dlouhodobou spoluprací se Západočeskou univerzitou v Plzni. Zároveň je firma Hobl&Pech s.r.o. a Západočeská univerzita v Plzni řešiteli vývojového projektu TA04031301 - Inteligentní elektronická stazka, který je dotován z dotačního programu TA ČR. V rámci tohoto projektu byla základní myšlenka IES ochráněna průmyslovým vzorem číslo 27 467 [37]. V roce 2015 je následně v plánu celou myšlenku a fungování IES ochránit patentem. Tento vývojový projekt je plánován do konce roku 2017 a vývoj systému Inteligentní Elektronické Stazky bude nadále pokračovat.

6. Závěr

Na počátku projektu Inteligentní Elektronická Stazka byla myšlenka vytvořit inteligentní a automatizovaný systém elektronického záznamu dopravních dat vozidel a strojů. Přesto, že na trhu již existují elektronické systémy nahrazující ručně psané záznamy, nejsou tyto systémy dostatečně chytré či automatizované. Proto bylo cílem vytvořit zcela nový systém, který nejen že obsluhu vozidla či stroje neobtěžuje, ale i přátelským způsobem ulehčuje práci, umožňuje digitální komunikaci, zpětnou vazbu s dispečinkem firmy a celkově přináší novodobý standard u dopravních společností.

Bakalářská práce autora se zaměřovala na automatizační jádro systému IES. Tato diplomová práce v první části popisuje systém IES jako celek, který obsahuje nejen samotné řidiče a jejich mobilní zařízení s IES, ale i propojení na dispečink firmy a životní cyklus informací relativních k dopravním procesům. Druhá část této práce se zaměřuje na přivedení určité inteligence do toho systému a to převážně multimodální ovládní, které tento systém vyzdvihuje v možnosti využití o třídu výše nad ostatní produkty na trhu.

Při implementaci syntézy řeči z textu byly otestovány technologie společnosti Google Inc. a SpeechTech s.r.o. Systém Google TTS nemohl být vhodně využit, jelikož nepodporuje český jazyk. U systému SpeechTech TTS byla otestována možnost syntézy s připojením k internetu a bez připojení. Vzhledem k požadavkům a výsledkům bylo zvoleno propojení IES a systému TTS společnosti SpeechTech pro operační systém Android. Implementace možnosti automatického rozpoznávání řeči započala definováním globálního slovníku příkazů pro IES. Dále byly vytvořeny sady testovacích nahrávek od několika řečníků při různých podmínkách. Tyto sady byly otestovány v systému Google ASR s gramatikou pro celý český jazyk a v systému ASR Západočeské univerzity při použití různých gramatik. Vzhledem k výsledkům testování byla zvolena možnost využívání systému ZČU ASR, u kterého je možné nastavit vlastní, zredukovanou gramatiku a tím velmi zvýšit úspěšnost rozpoznávání při ruchu z dopravního provozu na pozadí. Dále je počítáno s vývojem systému ZČU ASR na systém Android, aby bylo možné rozpoznávat příkazy pro IES bez připojení k internetu.

Vývoj tohoto produktu bude pokračovat i po této práci a to v rámci projektu TA04031301. Firma Hobl&Pech s.r.o. chce docílit toho, aby na trh přinesla nejen produkt schopný předčít konkurenci, ale produkt, který zcela změní dosavadní možnosti elektronických dopravních záznamů a vytvoří nový standard pro dopravní firmy.

7. Literatura a reference

- [1] **Hobl & Pech s.r.o.:** www.hoblapech.cz
- [2] **BRATRŠOVSKÝ, P., Automatizace získávání dopravních dat s využitím geolokace, 2013:**
theses.cz/id/e27gus
- [3] **Západočeská univerzita v Plzni:** www.zcu.cz
- [4] **Projet TA ČR č. TA04031301, 2014:**
www.isvav.cz/projectDetail.do;jsessionid=0D52DE7D6B222D5C039A1C428E7822C9?rowId=TA04031301
- [5] **GPS, 2015, [cit. 2.1.2015]:** cs.wikipedia.org/wiki/GPS
- [6] **Android, 2014, [cit. 18.1.2015]:** www.android.com
- [7] **Bluetooth, 2015, [cit. 20.1.2015]:** cs.wikipedia.org/wiki/Bluetooth
- [8] **DMD, 2015, [cit. 22.1.2015]:** www.hoblapech.com/cs/produkty
- [9] **FTP, 2015, [cit. 28.1.2015] :** cs.wikipedia.org/wiki/File_Transfer_Protocol
- [10] **Princip a.s., 2015:** www.princip.cz
- [11] **GX Solutions Bohemia s.r.o., 2015:** www.gxsolutions.cz
- [12] **CAN, 2015, [cit. 3.2.2015] :**
www.iae.fme.vutbr.cz/userfiles/beran/files/Datová%20sběrnice%20CAN.pdf
- [13] **FMS, 2015, [cit. 3.2.2015]:** <http://www.fms-standard.com/Truck/index.htm>
- [14] **Bluetooth Low Energy BLE, 2015:**
developer.android.com/guide/topics/connectivity/bluetooth-le.html
- [15] **GRPS, 2015:** cs.wikipedia.org/wiki/General_Packet_Radio_Service
- [16] **AETR, 2012, [cit. 10.2.2015]:** doprava.vpraxi.cz/aetr.html
- [17] **JSON:** www.json.org
- [18] **noSQL, 2014:** cs.wikipedia.org/wiki/NoSQL
- [19] **Couchbase, 2015, [cit. 19.2.2015]:** www.couchbase.com
- [20] **HTTP, 2014:** cs.wikipedia.org/wiki/Hypertext_Transfer_Protocol
- [21] **WebSocket, 2013:** www.websocket.org
- [22] **API, 2015, [cit. 27.2.2015]:** cs.wikipedia.org/wiki/API

- [23] TTS, 2015, [cit. 5.3.2015]: en.wikipedia.org/wiki/Speech_synthesis
- [24] MATOUŠEK, J., **Počítačová syntéza řeči, 2008, [cit. 7.3.2015]:**
www.isvav.cz/resultDetail.do;jsessionid=8D208196837B385551F51D5A45751D1A?roleId=RIV%2F49777513%3A23520%2F08%3A00502148!RIV10-MSM-23520___
- [25] HMM, 2015, [cit. 16.3.2015]: en.wikipedia.org/wiki/Hidden_Markov_model
- [26] ASR, 2015, [cit. 21.3.2015]: en.wikipedia.org/wiki/Speech_recognition
- [27] KRYŠKE, L., **Rozpoznávání řeči s pomocí nástroje SPHINX-4, 2014, [cit. 19.3.2015]:**
www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=86497
- [28] LVCSR, 2015, [cit. 24.3.2015]: en.wikipedia.org/wiki/Speech_analytics#LVCSR_.28large-vocabulary_continuous_speech_recognition.29
- [29] SpeechTech s.r.o.: www.speechtech.cz
- [30] Google Inc.: www.google.com
- [31] Android API, 2015: developer.android.com/guide/topics/manifest/uses-sdk-element.html
- [32] Flac, 2015: cs.wikipedia.org/wiki/Free_Lossless_Audio_Codec
- [33] ESGF, 2008, [cit. 12.4.2015]: voice.zcu.cz/VoiceXML/download/doc/esgf.pdf
- [34] 0-Gram LM, 2014, [cit. 15.4.2015]: www.cs.upc.edu/~gatus/mai-inlp/languagemodel2.pdf
- [35] Google Speech API, 2014: github.com/gillesdemey/google-speech-v2
- [36] JAVA: www.java.com
- [37] **Systém elektronické evidence dopravních údajů, 2014, [cit. 1.5.2015]:**
www.isvav.cz/resultDetail.do;jsessionid=CF7AF1D7F624E009F432F6B0ADA51E5E?roleId=RIV%2F49192787%3A_____%2F14%3A%230000003!RIV15-TA0-49192787
- [38] PSUTKA, J., MÜLLER, L., MATOUŠEK, J., RADOVÁ, V. **Mluvíme s počítačem česky. Academia, Praha, 2006, [cit. 5.4.2015]:**
www.kky.zcu.cz/cs/publications/PsutkaJ_2006_Mluvimes
- [39] FREDERICK, J., **Statistical Methods for Speech Recognition, 1997, [cit. 2.4.2015]:**
books.google.cz/books/about/Statistical_Methods_for_Speech_Recogniti.html?id=1C9dzcJTWowC&hl=cs