

Vývoj samoříditelné platformy

Filip Peterek

Vysoká škola Báňská - Technická univerzita Ostrava

Květen 2021

Obsah

Cíl práce

Výstup práce

Architektura projektu

Komponenty projektu

Vývoj projektu a testování způsobů řízení

Zhodnocení práce

Cíl práce

- ▶ Prozkoumat možnosti autonomního řízení vozidla
- ▶ Implementovat autonomní řízení v kampusu VŠB-TUO
- ▶ Implementovat komunikaci s vozidlem pomocí rozhraní CAN
- ▶ Implementovat simulátor vozidla sloužící k testování SW

Obsah

Cíl práce

Výstup práce

Architektura projektu

Komponenty projektu

Vývoj projektu a testování způsobů řízení

Zhodnocení práce

Výstup práce

- ▶ Následování grafických kódů
- ▶ Řízení pomocí GPS souřadnic
- ▶ Rozpracované řízení využívající vstupů z kamer, LiDARu i GPS
- ▶ Webová aplikace sloužící k ovládání vozidla
- ▶ Python skript sloužící ke stahování a exportu mapových podkladů
- ▶ Simulátor vozidla
- ▶ Komunikace s vozidlem za využití rozhraní CAN
- ▶ Android aplikace logující GPS souřadnice

Obsah

Cíl práce

Výstup práce

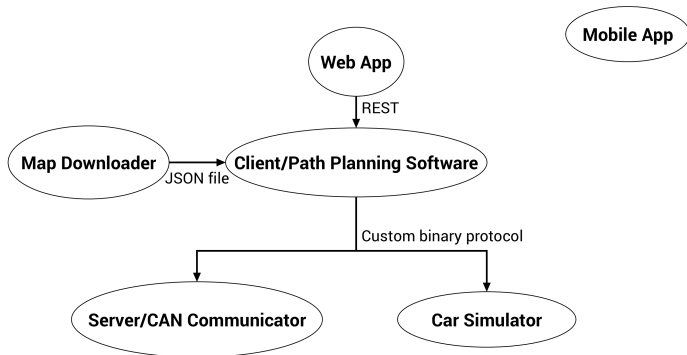
Architektura projektu

Komponenty projektu

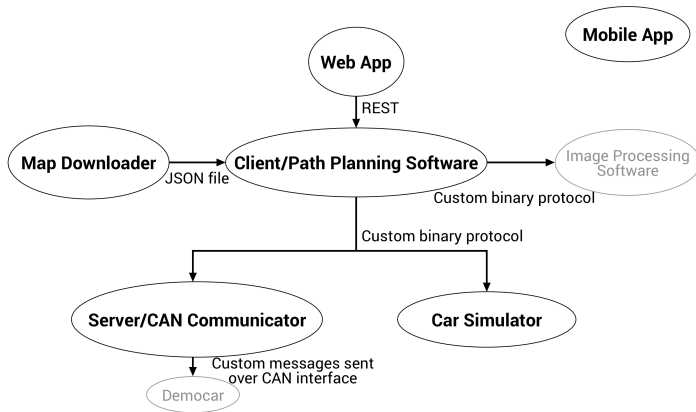
Vývoj projektu a testování způsobů řízení

Zhodnocení práce

Architektura projektu



Architektura projektu



Obsah

Cíl práce

Výstup práce

Architektura projektu

Komponenty projektu

Vývoj projektu a testování způsobů řízení

Zhodnocení práce

- ▶ Python 3.8
- ▶ python-can
- ▶ Abstrakce komunikace přes CAN rozhraní
- ▶ Pomocí TCP rozhraní umožňuje předávat vozidlu pokyny

car-simulator

- ▶ Python 3.8
- ▶ Jednoduchá fyzika – vozidlo je pouze hmotný bod
- ▶ Poskytuje stejné rozhraní jako CAN komunikátor
- ▶ Slouží především k jednoduchému testování

car-client

- ▶ Python 3.8
- ▶ Slouží k plánování cesty
- ▶ Komunikuje se serverem na vozidle
- ▶ Následování grafického bodu
- ▶ Následování GPS souřadnic
- ▶ Implementuje REST API umožňující zadání cíle cesty
- ▶ Práce s mapovými podklady

car-webapp

- ▶ Kotlin
- ▶ Mapy.cz
- ▶ REST API
- ▶ Webová aplikace
- ▶ Backend komunikuje s REST API plánovacího softwaru
- ▶ Backend validuje vstup – veřejné API
- ▶ Umožňuje zadávat cíle cesty vozidla
- ▶ Umožňuje sledovat pozici vozidla

car-map-downloader

- ▶ Python 3.8
- ▶ Stahuje mapové podklady OSM
- ▶ Export dat do vlastního JSON formátu
- ▶ Lze naschedulovat cronem – periodická obnova dat

GeoLogger

- ▶ Kotlin
- ▶ Jednoduchá Android aplikace
- ▶ Loguje GPS souřadnice do souboru
- ▶ Sloužila k testování přesnosti GPS

Obsah

Cíl práce

Výstup práce

Architektura projektu

Komponenty projektu

Vývoj projektu a testování způsobů řízení

Zhodnocení práce

Vývoj projektu

- ▶ Několik verzí CAN specifikace
- ▶ Několik způsobů řízení
 - ▶ Řízení pomocí GPS
 - ▶ Řízení pomocí detekce grafických kódů
 - ▶ Řízení pomocí kombinace vstupů z kamer, GPS, LiDARu

Řízení pomocí GPS

- ▶ Vozidlo jede rovně k nejbližšímu waypointu
- ▶ Řízeno pouze pomocí GPS
- ▶ Vysoká nepřesnost GPS
- ▶ Aplikace filtru pomohla, ale nedostatečně

Řízení pomocí detekce grafických kódů

- ▶ Pomocí kamery je detekován kód
- ▶ Plánovací software pozici kódu získá přes TCP rozhraní
- ▶ Vozidlo je vedeno přímo ke kódu
- ▶ Funkční, ale nepraktické
- ▶ Problém s fyzickým rozmístěním kódů

Řízení pomocí vstupů z kamer, GPS, LiDARu

- ▶ Vozidlo ledá cestu v mapových podkladech
- ▶ Danou cestu následuje
- ▶ Bezpečnost provozu, dodržování směru, atd. zajištěna kamerami
- ▶ Momentálně ve vývoji

Dopad koronaviru

- ▶ Znemožněno testování na fyzickém vozidle
- ▶ Nedostatek HW
- ▶ Plánovací software opožděn
- ▶ Čas byl využit k implementaci webové aplikace, obnově mapových podkladů

Obsah

Cíl práce

Výstup práce

Architektura projektu

Komponenty projektu

Vývoj projektu a testování způsobů řízení

Zhodnocení práce

Zhodnocení práce

- ▶ Vyzkoušeny dva způsoby řízení
- ▶ Funkční komunikace přes rozhraní CAN
- ▶ Vzdálené ovládání vozidla pomocí TCP rozhraní
- ▶ Webová aplikace
- ▶ Základ řízení pomocí kombinace vstupů
- ▶ Vývoj bude pokračovat

Konec

Děkuji za pozornost