**TEHNIČNA DOKUMENTACIJA – Carobotics**

Sestavni deli projekta:

* Hub (vozlišče med algoritmi in simulatorjem)
* AI (program za avtonomno vožnjo vozila)
* AirSim (vizualizacija parametrov vožnje)
* STM program za manualno kontrolo avtomobila
* STM - Hub vmesnik za pošiljanje podatkov

Programski jeziki:

* Python
* Golang
* C
* C#

Operacijski sistem: macOS Big Sur

1. **Hub**Programske zahteve:  
   - Python (verzija 3.7)  
   Knjižnjice**:**- airsim  
   - cv2  
   - numpy  
   - os  
   - base64  
   - json  
   - time  
   - flask  
   - decimal  
   - functools  
     
   Hub nam služi kot centralno vozlišče med algoritmi in AirSim, ki teče znotraj Unity.   
   Vozlišče je implementiramo z Flask framework-om in deluje kot server na katerega kličemo GET in POST zahteve.  
   Preko vozlišča lahko dobimo vse podatke iz simulatorja (hitrost, usmerjenost koles, pospešek, zaviranje, ročna zavora, slika iz kemere, lokacija avtomobila, podatki iz IMU senzorja, informacije o trku vozila, podatek kdo upravlja vozilo (algoritem, človek)).  
   Centralno vozlišče nam tudi omogoča kontrolo nad vozilom, kjer lahko sami nastavljamo parametre vožnje (pospeševanje, zaviranje, usmerjenost koles, ročna zavora). Za normalno delovanje moramo za  
   Vozlišče poženemo z ukazom: *python3 hub.py*
2. **AI**

Programske zahteve:  
- Python  
  
Knjižnjice:  
- numpy  
- requests  
- time  
- cv2  
- base64  
- io  
- datetime  
  
Program preko centralnega vozlišča z GET zahtevo prejme sliko iz avtomobila.  
Slika se zatem pretvori v HSV barvni prostor. Vse pixle, ki niso v območju za katerega smo ocenili, da predstavlja rdečo barvo, nastavimo na vrednost 0. Vse pixle, ki pa so bili zaznani kot rdeča barva pa nastavimo na vrednost 255. Na sliki je tako vidna samo zaščitna ograja.   
Zavijanje v levo ali desno nastavimo glede na podatek na kateri polovici slike (leva ali desna) je več pixlov, ki predstavljajo zaščitno ograjo. Podatek o usmerjenosti koles se na koncu pošlje na centralno vozlišče od koder se upravlja avtomobil.  
Pogoj za normalno delovanje je delujoče centralno vozlišče.  
Program poženemo z ukazom: *python3 main.py*

1. **AirSim**  
   Programske zahteve:  
   - Unity Hub  
   - Unity (verzija 2019.3.12f1)  
   - AirSim (verzija #TODO)

V podmapo UnityDemo znotraj mape projekta AirSim skopiramo našo mapo Assets v kateri so vsi objekti, ki se bodo pojavili v sceni in skripte (C#) za nadzor upodabljanja vizualizacije nad temi objekti. Nato v Unity uvozimo AirSim projekt in ga zaženemo. Po končanem nalaganju pristisnemo gumb Play in v začetnem meniju izberemo Car Mode. Unity nam naloži začetno sceno in lahko pričnemo z vožnjo.

1. **STM**Programske zahteve:  
   - Cube IDE (1.5.0)  
   - C  
     
   Strojna oprema:  
   - STM F411E  
   - micro USB kabel  
   - mini USB kabel  
     
   Na STM F411E ploščico z razvijalnim orodjem Cube IDE naložimo program, ki iz senzorja pridobiva podatke o nagibu in jih posreduje računalniku preko USB povezave. Uporabnik ob lihem pritisku na gumb sam upravlja z vozilom z nagibanjem ploščice. Ob sodem pritisku nadzor nad vozilom prevzame AI program.  
   Na ploščici so implementirana opravila (branje gumba, pošiljanje podatkov gumba, branje senzorja, pošiljanje podatkov senzorja), ki se glede na zahteve RTOS izvajajo v določenem vrtsnem redu.  
   Za normalno delovanje moramo ploščico priključiti na računalnik z obema kabloma.
2. **STM – Hub vmesnik**Opomba: dela le na macOS Big SurProgramske zahteve:  
   - Golang  
     
   Strojna oprema:  
   - STM F411E  
   - micro USB kabel  
   - mini USB kabel  
     
   STM F411E kot CDC naprav preko USB povezave pošilja podatke o nagibu ploščice na računalnik. Znotraj vmesnika te podatke sprocesiramo in jih posredujemo našemu kontrolnemu vozlišču (POST zahteva).  
   Vmesnik zaženemo z naslednjim ukazom: *./CDCController XXX YYY*  
     
   XXX = številka naprave, ki jo dobimo z ukazom: *ls /dev/tty\**YYY = naslov na katerem teče cebtralno vozlišče (privzeto *http://127.0.0.1:5000*)  
     
   Primer zagona STM – Hub vmesnika:   
   *./CDCController /dev/tty.usbmodem3496364D33371* *http://127.0.0.1:5000*  
     
    **VRSTNI RED UKAZOV:**- zagon AirSim projekta v Unity- *python3 hub.py* (centralno vozlišče - Hub)  
   - *./CDCController XXX YYY* (STM – Hub vmesnik, podroben opis ukaza razložen zgoraj)  
   - *python3 main.py* (AI program)