F1/10 YellowTails

User Manual

Bowen Boyd
Hanyue Wang
Kyle Watson
Jordan Wright

Faculty Mentor:
Isaac Shaffer

Project Sponsor:
Truong X. Nghiem, Assistant Professor, SICCS, NAU
Trong-Doan Nguyen, PhD Student, SICCS, NAU

05/01/2020

Table of Contents

1. Introduction

We are pleased that you have chosen, for your summer camp program, a highly interactive graphical user interface that has been custom-designed to meet your needs. Some of the key highlights of the product, RosConnect, include: profile load and save, dynamically configurable YAML file that determines the configuration window, save session feature, simulator option, and logging module. The purpose of this user manual is to help you, the client, successfully install, administer, and maintain RosConnect in your actual summer camp program context going forward. Our aim is to make sure that you are able to benefit from our product for many years to come!

2. Installation

As part of final delivery, the system should have been installed on a Raspberry Pi. Over time, however, you may want to move to a new platform or re-install RosConnect. To do so, we direct you to the following set of instructions for a complete installation.

Installing
A step by step of examples that tell you how to get the software running.

For this software to work ROS must be installed on the host machine. To learn how to install ROS visit ROS.org

Clone the repo into the home directory

```
cd ~/ && git clone https://github.com/jaw566/RosConnect.git
```

Run the setup script

```
cd ~/RosConnect/App/ && ./setup.sh [host_password] [vehicle_domain_name]
[vehicle_hostname] [vehicle_password]
```

Simulator Setup
In order to run the simulators you need to install and source the simulator.

We need the ros-kinetic-map-server to run the simulator

```
sudo apt-get install ros-kinetic-map-server
```

Make a new workspace for the simulator

```
Mkdir -p ~/f110_ros/src
cd ~/f110_ros/src
catkin_init_workspace
```

Clone the simulator

```
cd ~/f110_ros/src && git clone
https://github.com/FF1RR-NAU-Spring-2020/ff1rr-2020-spring.git
```

Make the workspace with catkin_make and source the file

```
cd ~/f110_ros/
catkin_make
source devel/setup.bash
```

To make the simulator work with out sourcing it every time add the source command to your bashrc file

```
echo 'source ~/ff110_ros/devel/setup.bash' >> ~/.bashrc
```

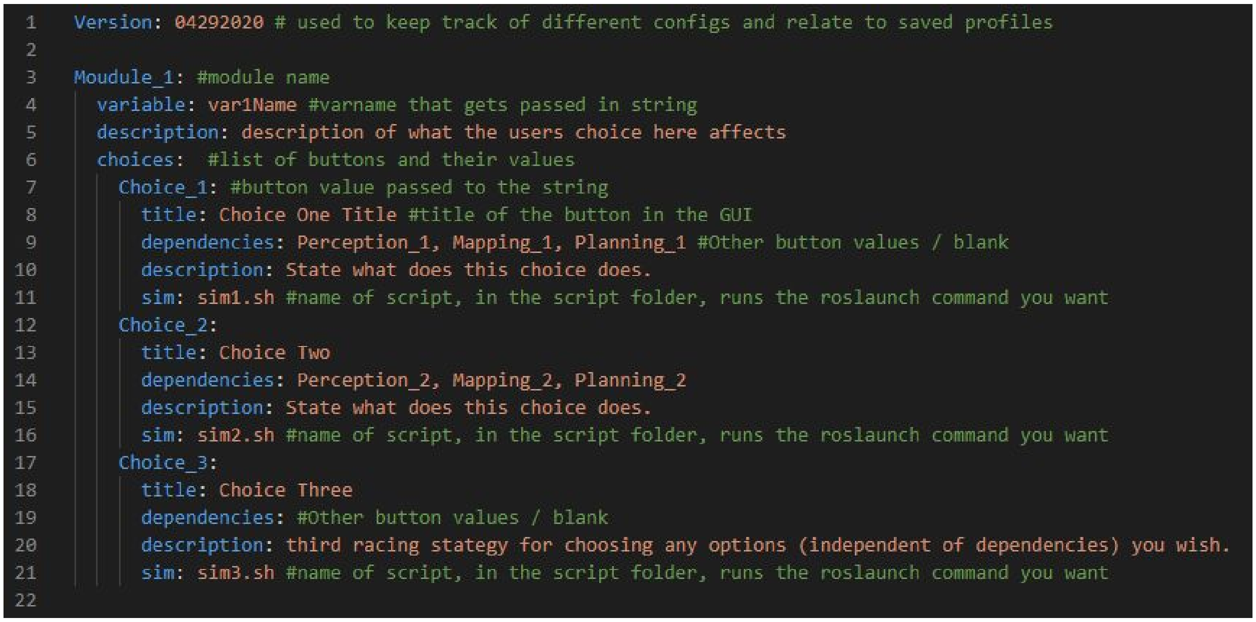
3. Configuration and Daily Operation

Daily operations must begin by ensuring that installation is up-to-date with any changes since the last time you installed RosConnect. Although the host name, host password, and vehicle password are set upon installation in Section 2, you may have changed these settings. If you have not changed these settings since your last install, then there is no need to reconfigure the setup settings of RosConnect. Otherwise, to configure the setup settings again, you will execute the following script in the App directory with your new passwords and/or host name as follows:

```
cd~/RosConnect/App/ && ./setup.sh [host_password] [vehicle_host_id] [vehicle_hostname]
[vehicle_password]
```

Making sure that you enter both vehicle_domain_name, vehicle_hostname and the vehicle_password, if you miss one you will get a warning telling you what to put in.

Now that the software is installed it needs to be configured to use with your current curriculum. The format of the config file is as follows:



```
Version: 04292020 # used to keep track of different configs and relate to saved profiles

Moudule_1: #module name
  variable: var1Name #varname that gets passed in string
  description: description of what the users choice here affects
  choices:  #list of buttons and their values
    Choice_1: #button value passed to the string
      title: Choice One Title #title of the button in the GUI
      dependencies: Perception_1, Mapping_1, Planning_1 #Other button values / blank
      description: State what does this choice does.
      sim: sim1.sh #name of script, in the script folder, runs the roslaunch command you want
    Choice_2:
      title: Choice Two
      dependencies: Perception_2, Mapping_2, Planning_2
      description: State what does this choice does.
      sim: sim2.sh #name of script, in the script folder, runs the roslaunch command you want
    Choice_3:
      title: Choice Three
      dependencies: #Other button values / blank
      description: third racing stategy for choosing any options (independent of dependencies) you wish.
      sim: sim3.sh #name of script, in the script folder, runs the roslaunch command you want

```

The rest of the modules follow the same format but without the sim variable.

The sim scripts should look like the following:

#!/bin/bash
#source $1

Roslaunch [ros_package] [rosFile.launch] && fg

The current values selected when users click “Start Car and Run Sim” follow the following format using the values that are set in the config file:

variable_name:=option_# variable_name:=option_# variable_name:=option_#
variable_name:=option_#

4. Maintenance

The maintenance of this product is fairly low. RosConnect maintenance is limited to updating all installation dependencies on an as-needed basis. That is, within the installation process, you will execute a setup script. In this setup script an installation of dependencies occurs by calling the command install.sh. This file exists in the App directory and contains install commands that install RosConnect dependencies with the most up-to-date versions. *If these versions should be updated in the future, then you must also update them in the install.sh script and reinstall RosConnect dependencies by either following directions outlined in Section 2 or directly running the install script.* That is, if you change dependency versions within the install commands in the file install.sh we direct you to either follow Section 2 directions for reinstalling or to execute the following command to reinstall dependencies for the RosConnect system.

Maintenance: Run the install dependencies script:
cd ~/RosConnect/App/ && ./install.sh [host_password]

5. **Trouble-Shooting**

If there are problems after making a new config try to delete the saved_data.txt file that is in the root directory. Also remember that saved profiles are dependent on the version number of the config so old saved profiles will not work with a new config file.

6. **Conclusion**

The idea that autonomous vehicle technology will be available to high school students and that our team has played a critical role in providing this unique opportunity has been both gratifying and fascinating. We hope that RosConnect will ensure you many happy years of productive summer camp programs! While we are all moving on to professional careers, we would be happy to answer short questions in the coming months to help you get the product deployed and operating optimally in your organization. So, if you have any questions, please email Kyle at kpw44@nau.edu. With best wishes from your developers: Bowen Boyd, Hanyue Wang, Kyle Watson, and Jordan Wright.