NAVIGATION AND TAG DETECTION FOR IOT APPLICATIONS

ROBOTICS PROJECT MSCV_2

Students: Agbohoui Olivier, Helmy Tawfik Mohamed, Kodipaka Vamshi, Legrand François

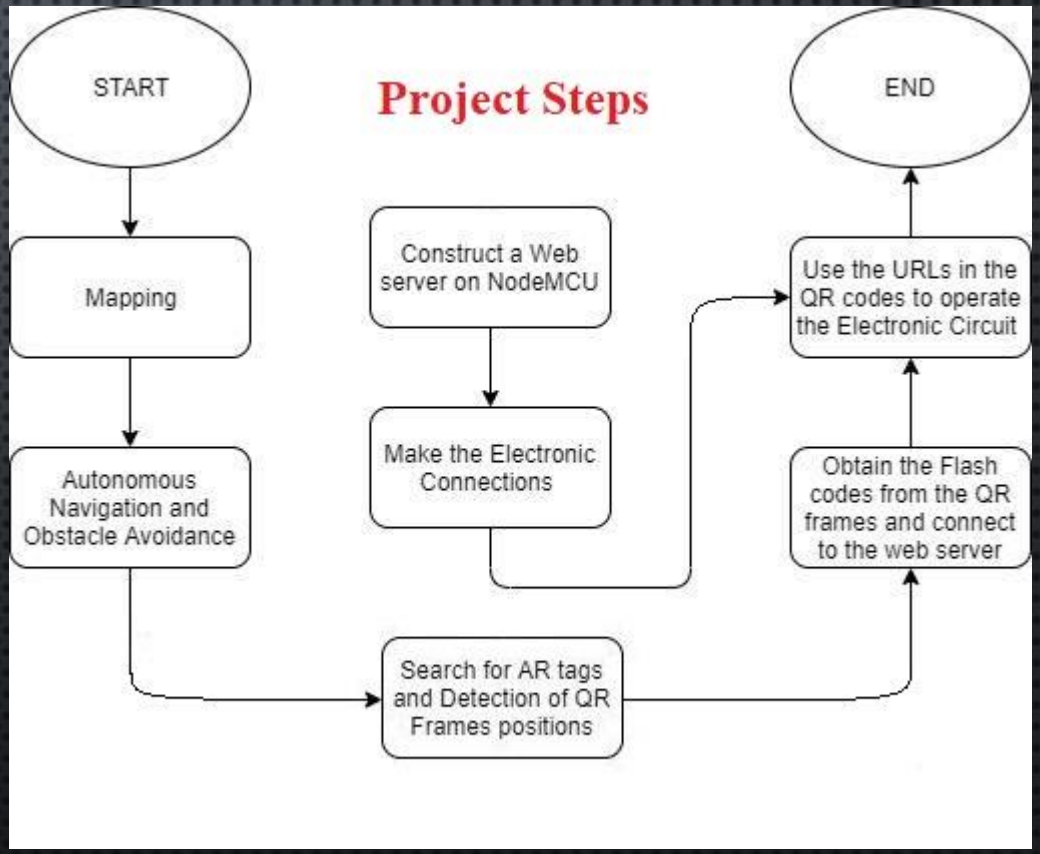
Supervized by : R. Seulin, M. Blanchon, D. Fofi, R. Duverne, D. Braun, T. Clamens

CONTENTS

I) – INTRODUCTION

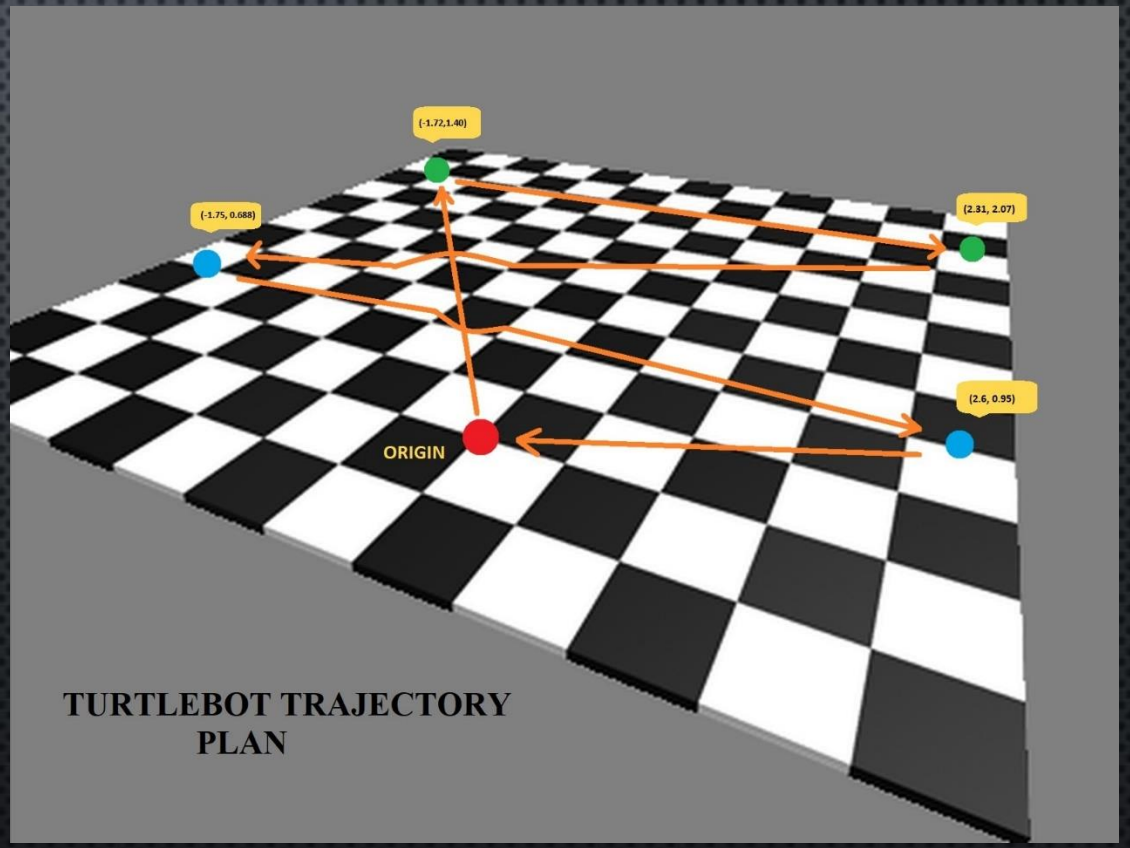
- Parts of the project

1. Autonomous Navigation
2. Vision tasks and basic detection
3. Wifi control of a system



I) - INTRODUCTION

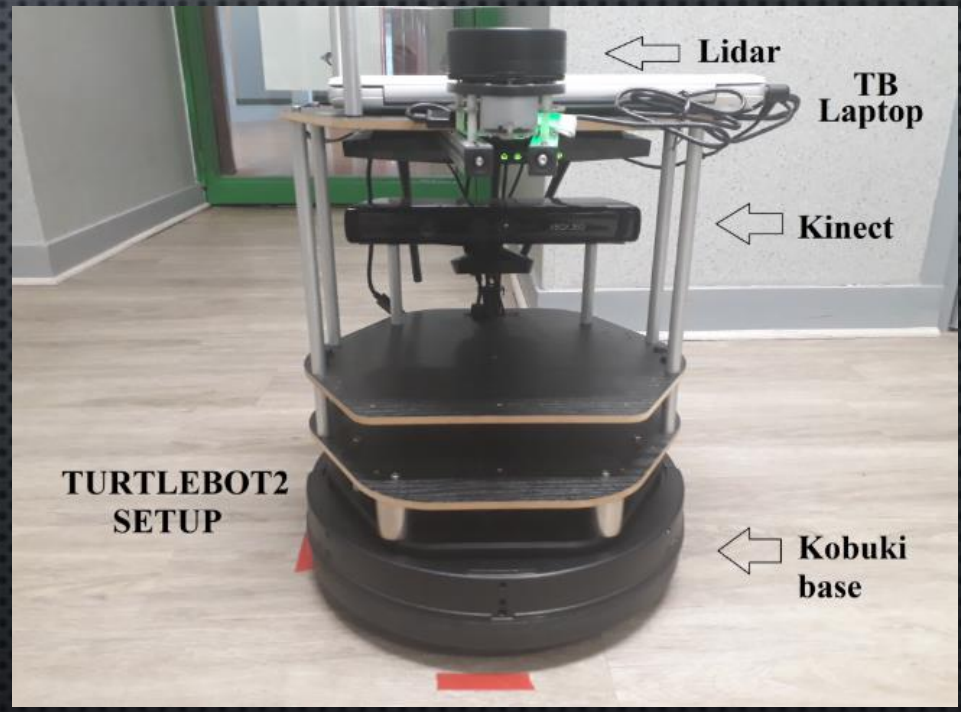
- Scenario
  1. Move to the AR Tags
  2. Move to the QR codes
  3. Remotely control the LED
  4. Go to the original position



II) – AUTONOMOUS NAVIGATION

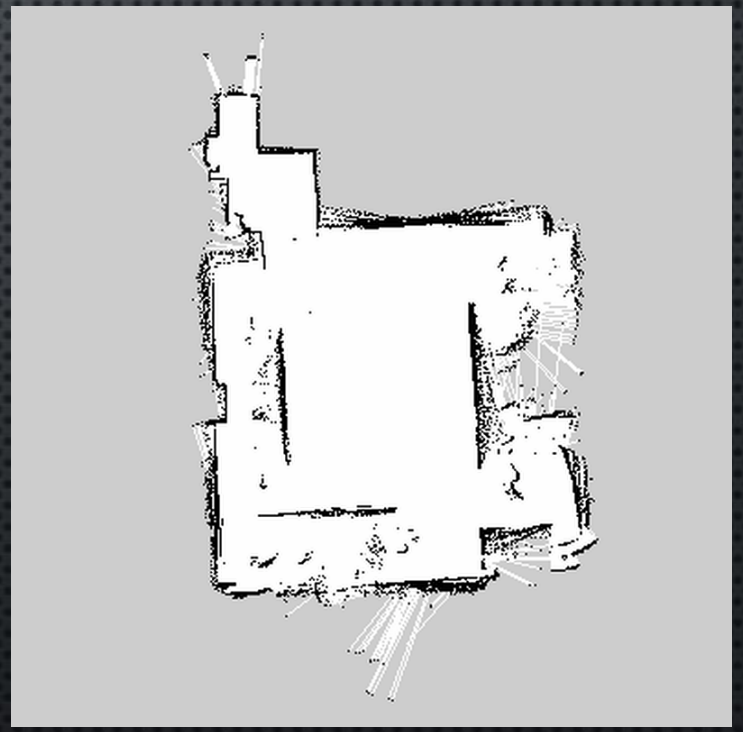
- Original Material

1. turtlebot_vibot metapackage
2. Slamtec package rplidar_ros
3. LiDAR scanner sensor



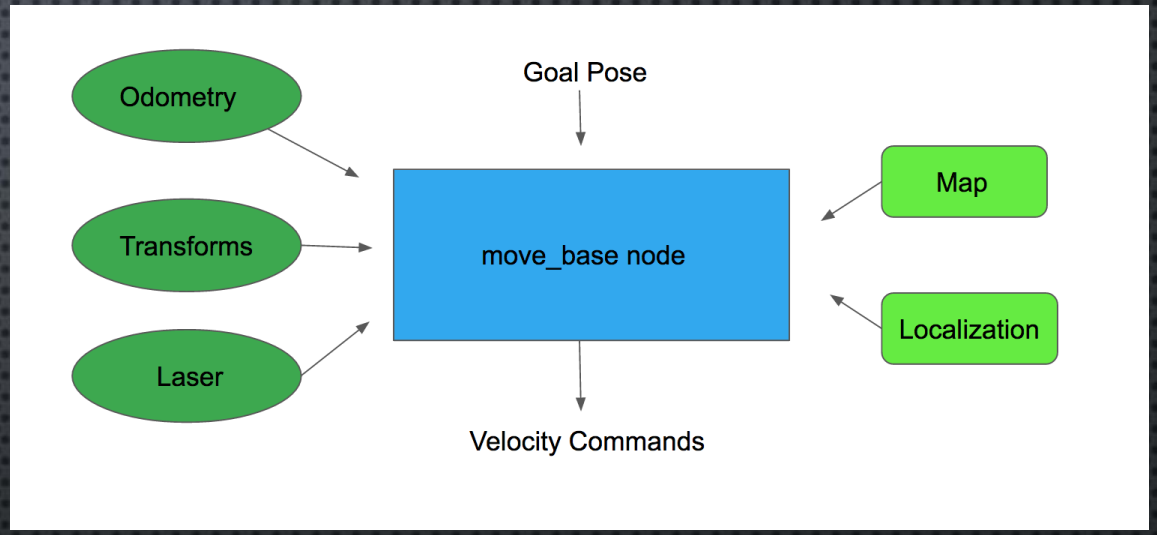
II) – AUTONOMOUS NAVIGATION

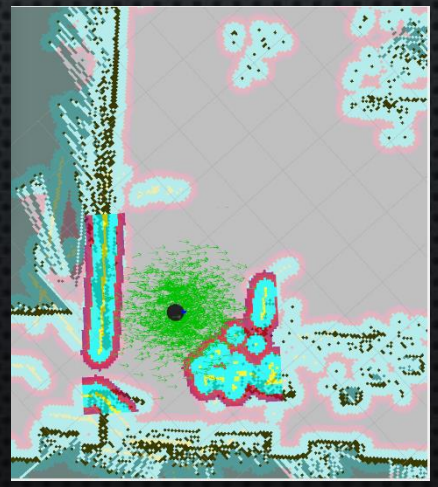
- Map Creation
  - Single time process
  - Pure LiDAR reliance
  - Use of gmapping_demo file
  - Manually controlled robot
  - Results stored into turtlebot_vibot_nav



II) – AUTONOMOUS NAVIGATION

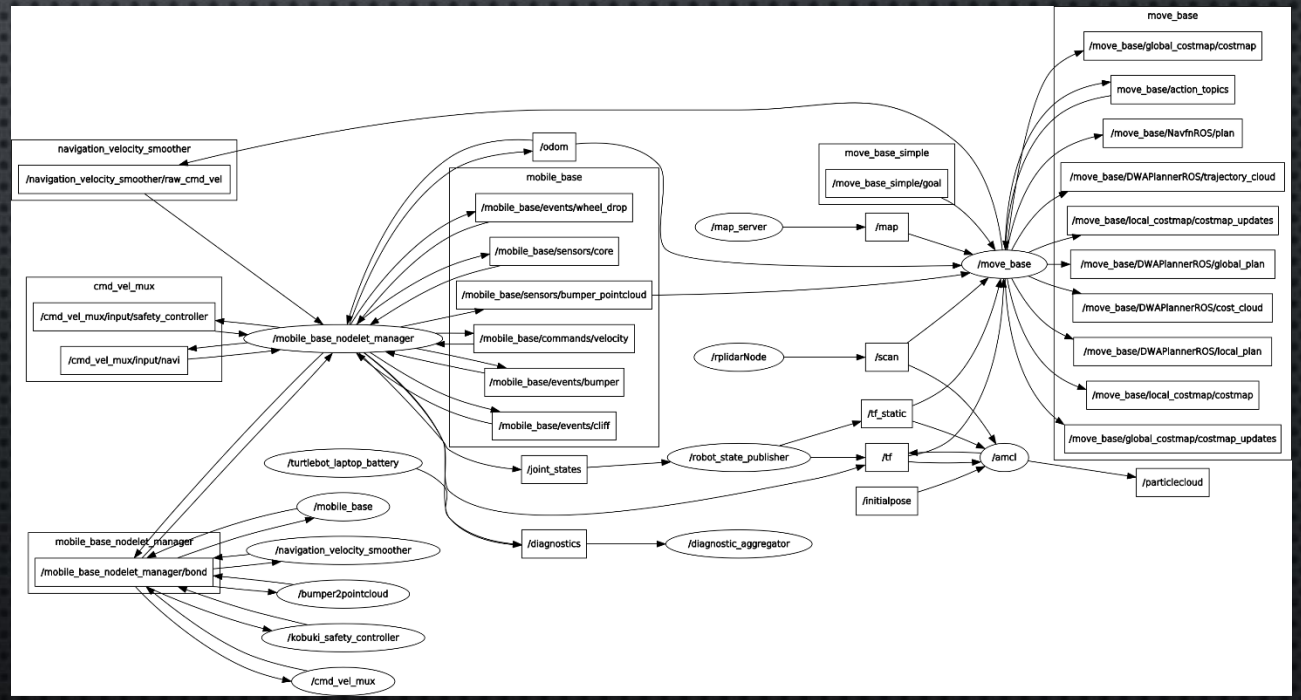
- Setting Autonomous navigation
  - LiDAR and Kinect dependent
  - Call of amcl_demo file
  - Real time local map
  - Exclusion radius for obstacle avoidance
  - Shortest path planning





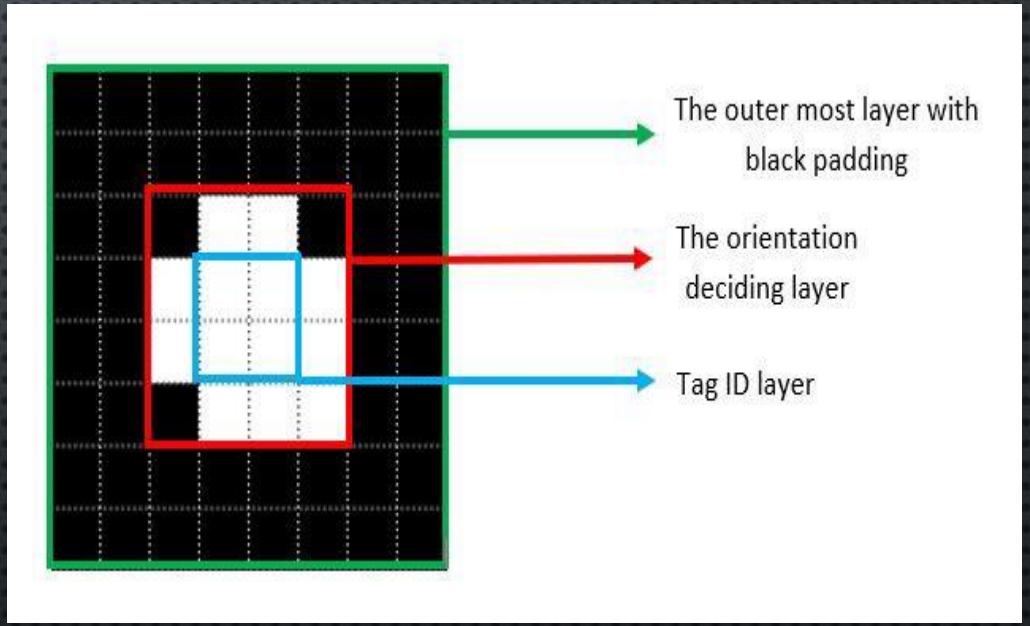
II) – AUTONOMOUS NAVIGATION

- Setting Autonomous navigation
  - Python script adjustment
  - Manually defined coordinates
  - Move_base node call



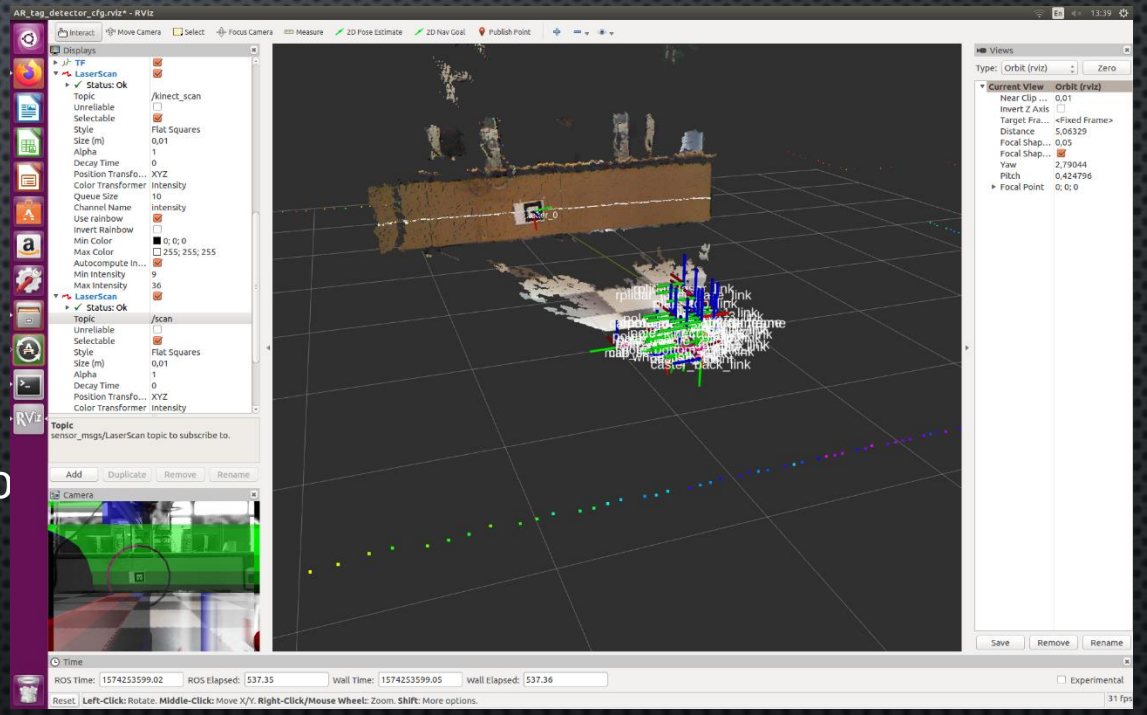
III) – AR TAGS LOCALIZATION / QR DETECTION

- AR_track_alvar adaptation
  - Remapping Kinect camera topic and framework
  - Generating basic simple AR Tags
  - Trying different setup parameters
  - Tag distance to the robot is passed into several topics



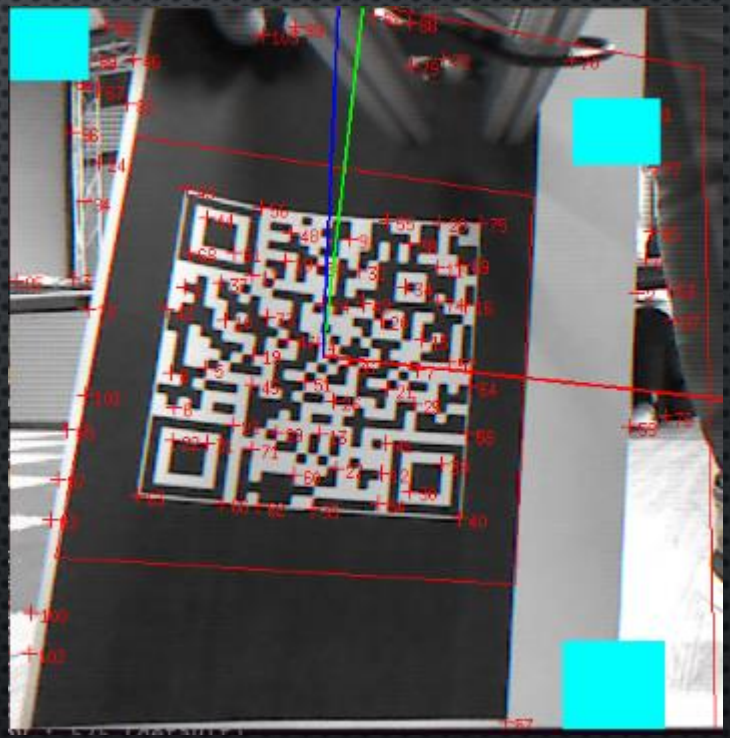
III) – AR TAGS LOCALIZATION / QR DETECTION

- AR_track_alvar adaptation
  - Output visualization through Rviz
  - Custom configuration for display
  - The detection subscribes to the cloud of points to
  - The orientation is also visible on camera output



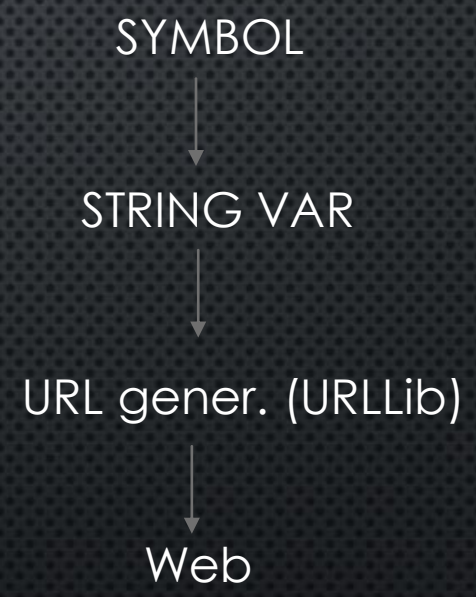
III) – AR TAGS LOCALIZATION / QR DETECTION

- Visp Auto-tracker Compatibility
  - Camera configs (launch files) share the same structure
  - Kinect camera is a USB camera
  - Direct remapping of the USB port camera to Kinect IDs
  - Detection based on fixed QR features
  - The detection takes into account orientation



III) – AR TAGS LOCALIZATION / QR DETECTION

- Zbar and pyzbar
  - QR and bar codes decoding/encoding libraries
  - Pyzbar unleashes all functionalities from Zbar in Python
  - Pre-processing is mandatory to reach optimal decoding
  - Decoding is done through embedded pattern scanning
  - Obtained symbols are encoded into a string variable
  - The resulting URL is processed through a URLLib call prior to web-browsing



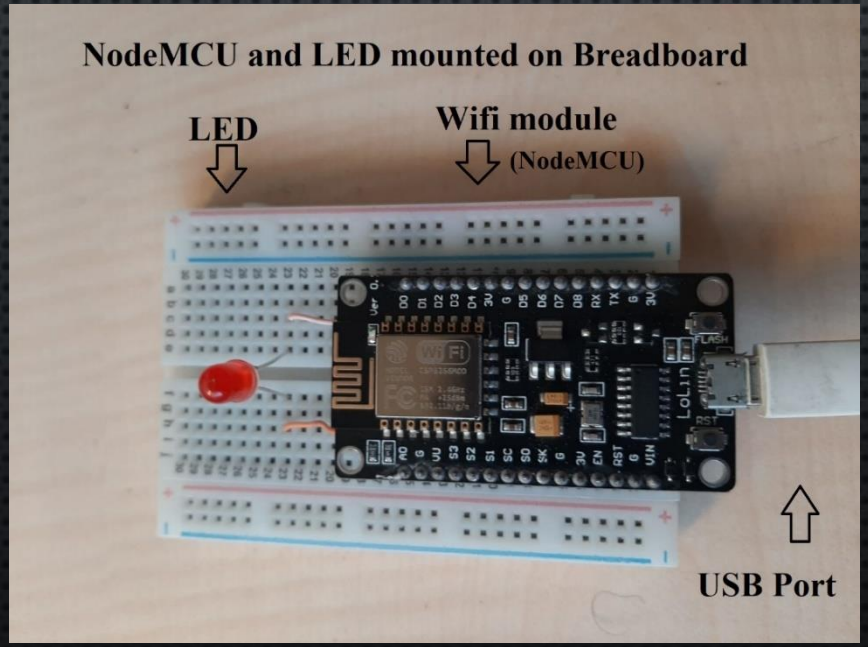
WI-FI CONTROL

IV) - WI-FI CONTROL

- Getting beyond ROS
  - Convert an image of a QR code into an Electrical signal
  - Use the Kinect to decode the QR Code in the image
  - Construct a webserver using a Wi-Fi Module embedded on the NodeMCU board.
  - Establish a connection between the ROS system and the webserver

IV) - WI-FI CONTROL

- Electronics
  - In order to achieve these objectives we used the NodeMCU
  - The webserver was constructed on the NodeMCU using Arduino C
  - The Electronic connection was made using a single LED

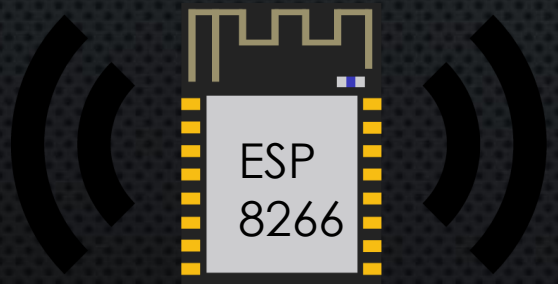


ARDUINO

+

webserver

=



PROCESS

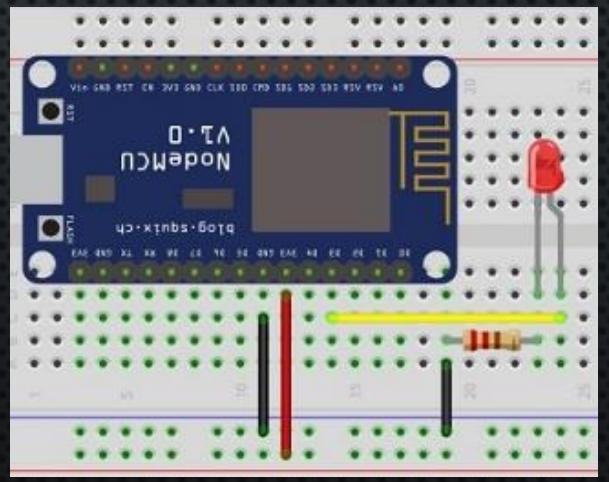
aREST Cloud Server

ID, Name, Configuration, Hardware

Kinect and QR Codes

Can only be done over a local network

NodeMCU Webserver



EXPERIMENT

- Below are the QR Codes used in the Project

Switch On

cloud.arest.io/mt1234/digital/3/1

Switch Off

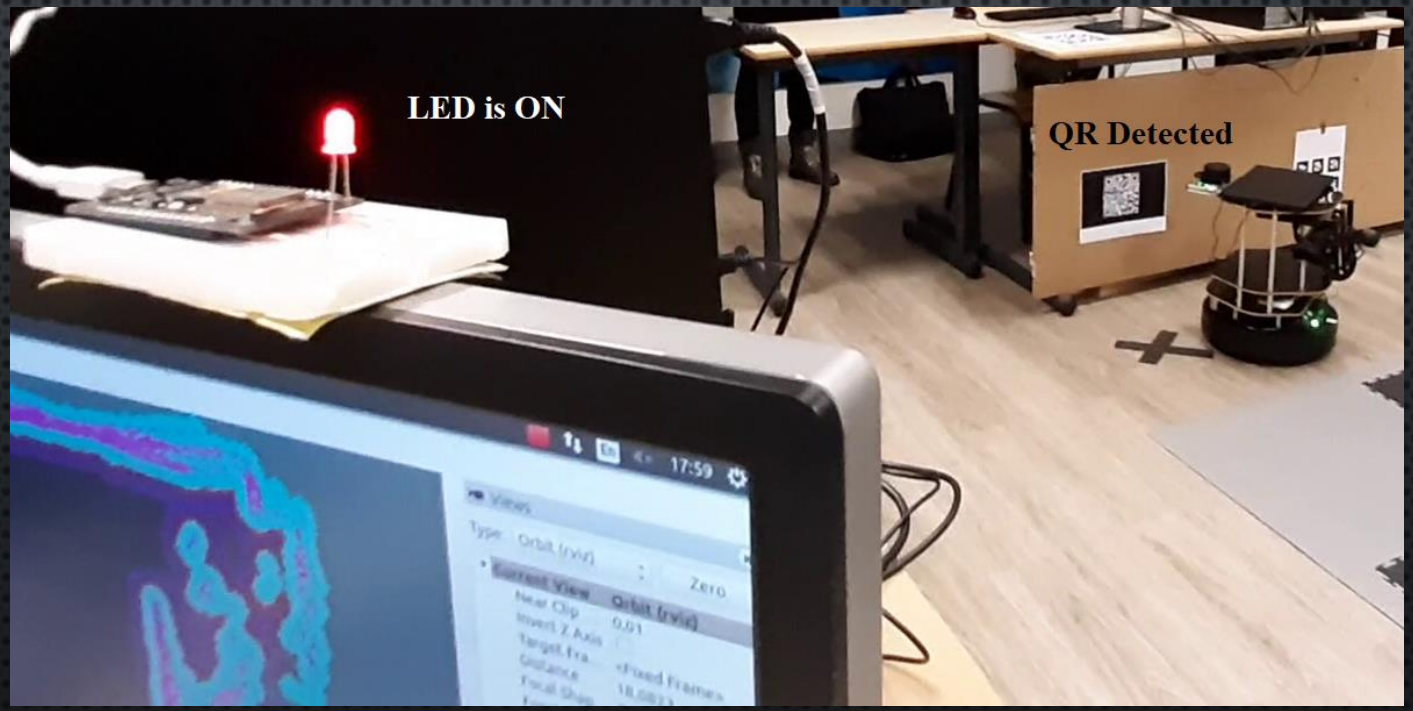
cloud.arest.io/mt1234/digital/3/0

V) APPLICATIONS

- Possible Expansions
  - State Machine temporal management with **Smach**
  - AR Tags for distance control
  - Connectivity with other robots
  - Gesture based navigation
- Real Life Uses
  - Light monitoring through displacement
  - Turning on a radio when inside a room
  - Adding smart sensors
  - Remote operating from long range

VI) CONCLUSION & DEMO

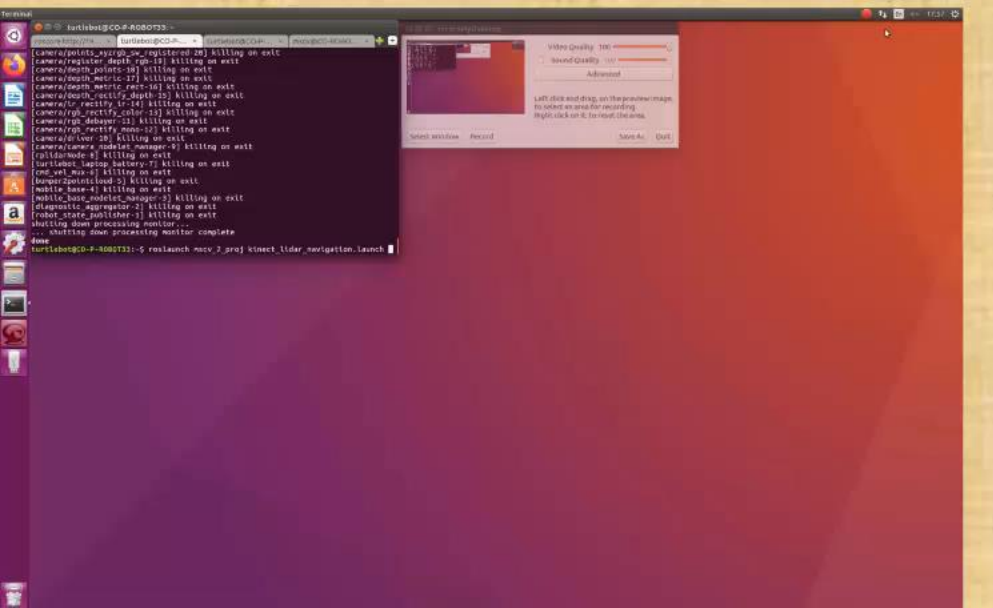
- External expansion to ROS
- Familiarization with the platform workflow



https://www.youtube.com/watch?v=rtxCUmQBTSE

MSCV2 ROS PROJECT

Navigation and Tag Detection for IoT Applications



THANKS FOR YOUR ATTENTION