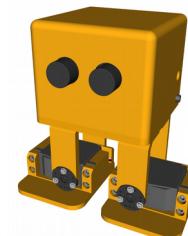




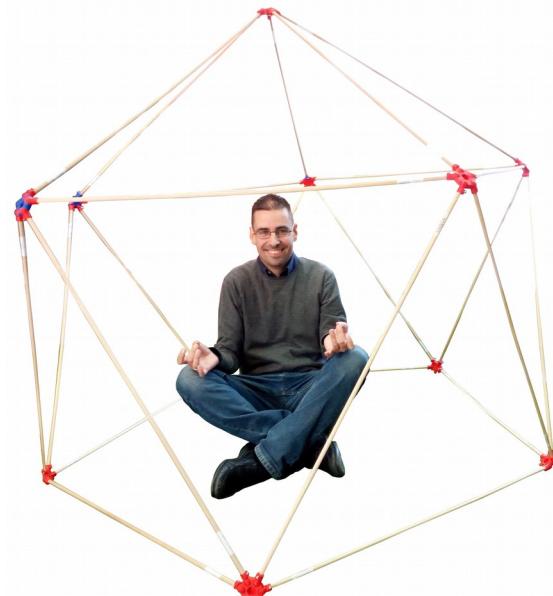
Entorno JdeRobot-Academy para la docencia práctica de robótica





Universidad
Rey Juan Carlos

Grupo de Robótica



Gsyc



Juan González Gómez
[@Obijuan_cube](https://twitter.com/Obijuan_cube)



online



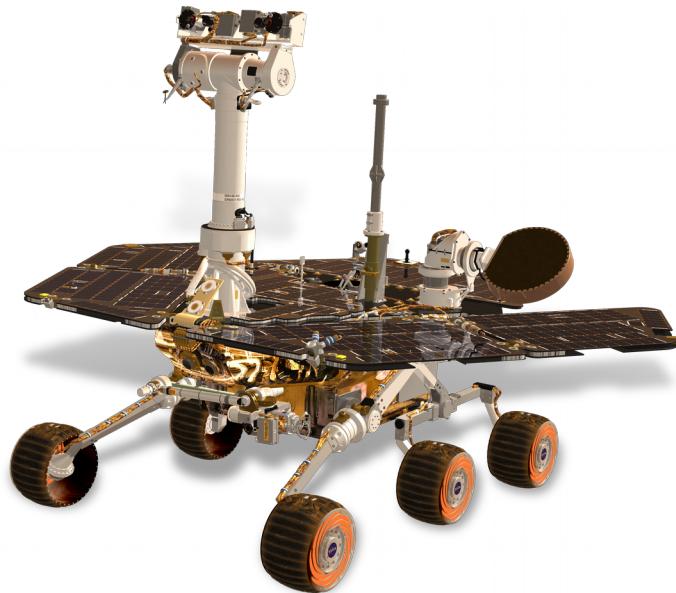
Robots en Ciencia-ficción



íñonline



Robótica en la vida real



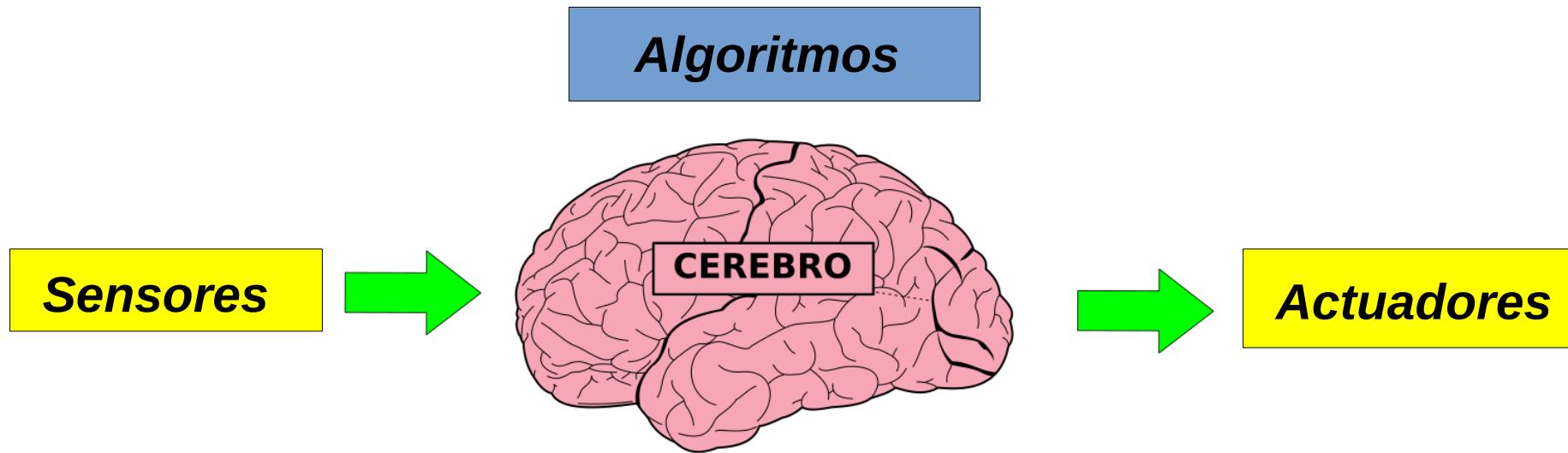
online

Robots móviles: campo prometedor



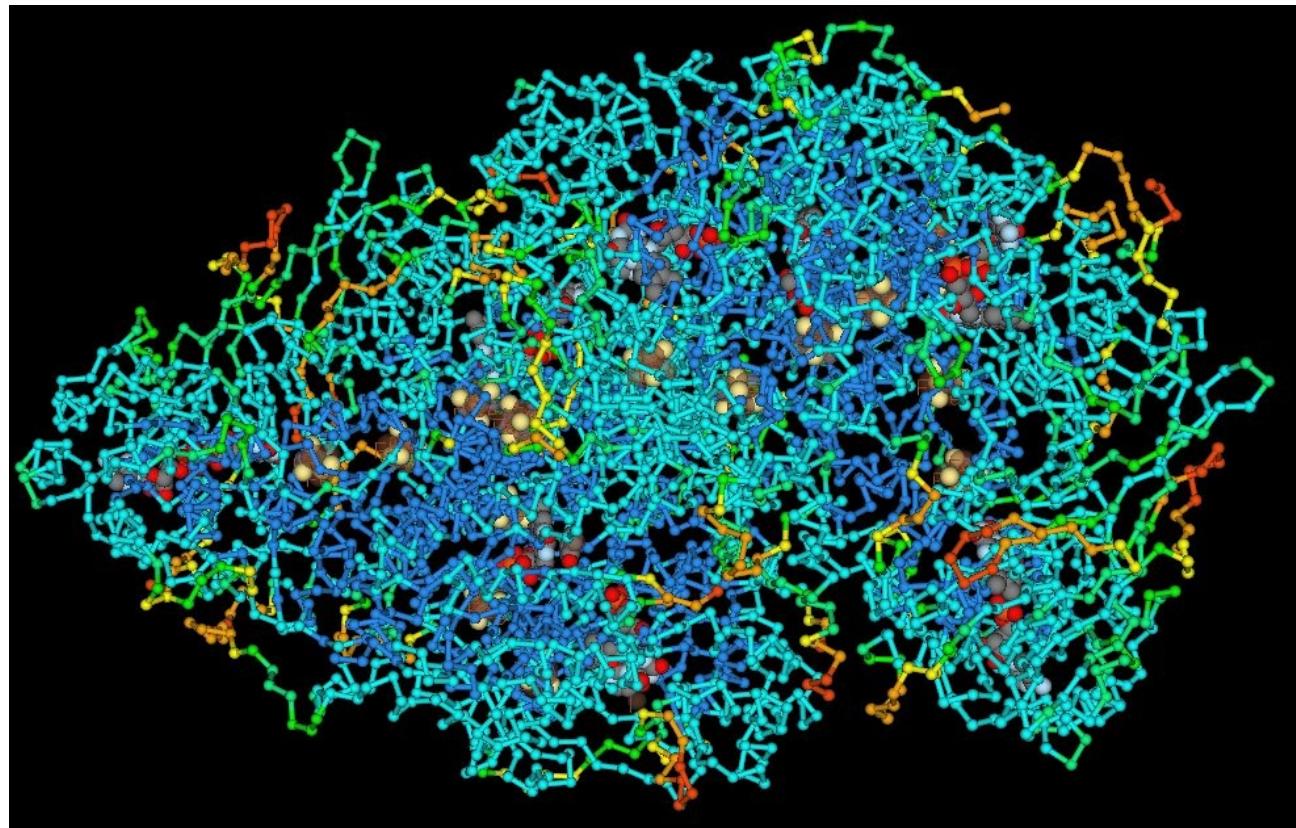
online

Énfasis en los algoritmos



online

Principios de diseño: Dosificación de la complejidad



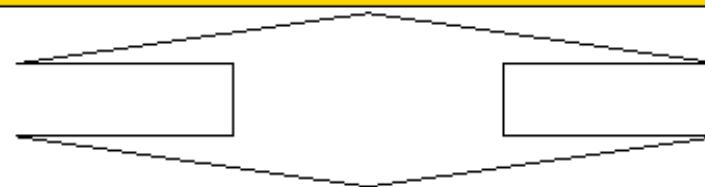
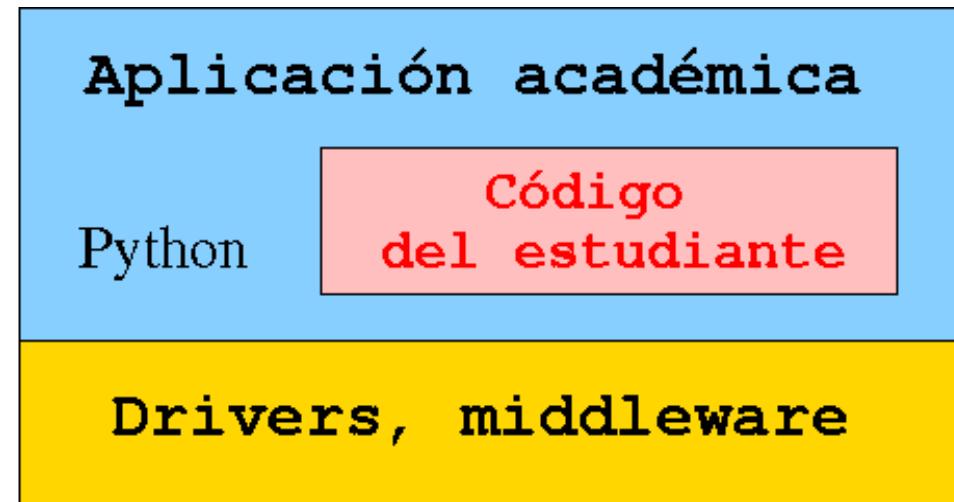
online

Solución: Aplicación académica para cada práctica

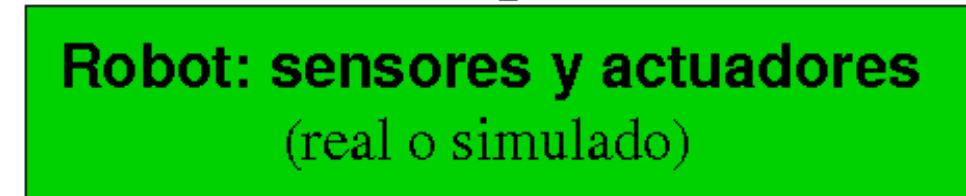


JdeRobot

SW

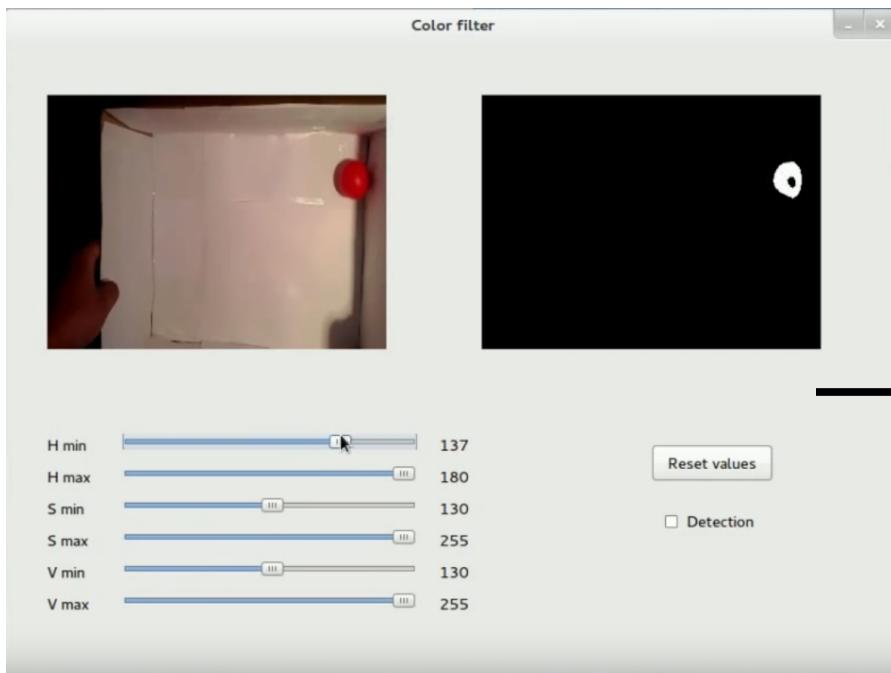


HW



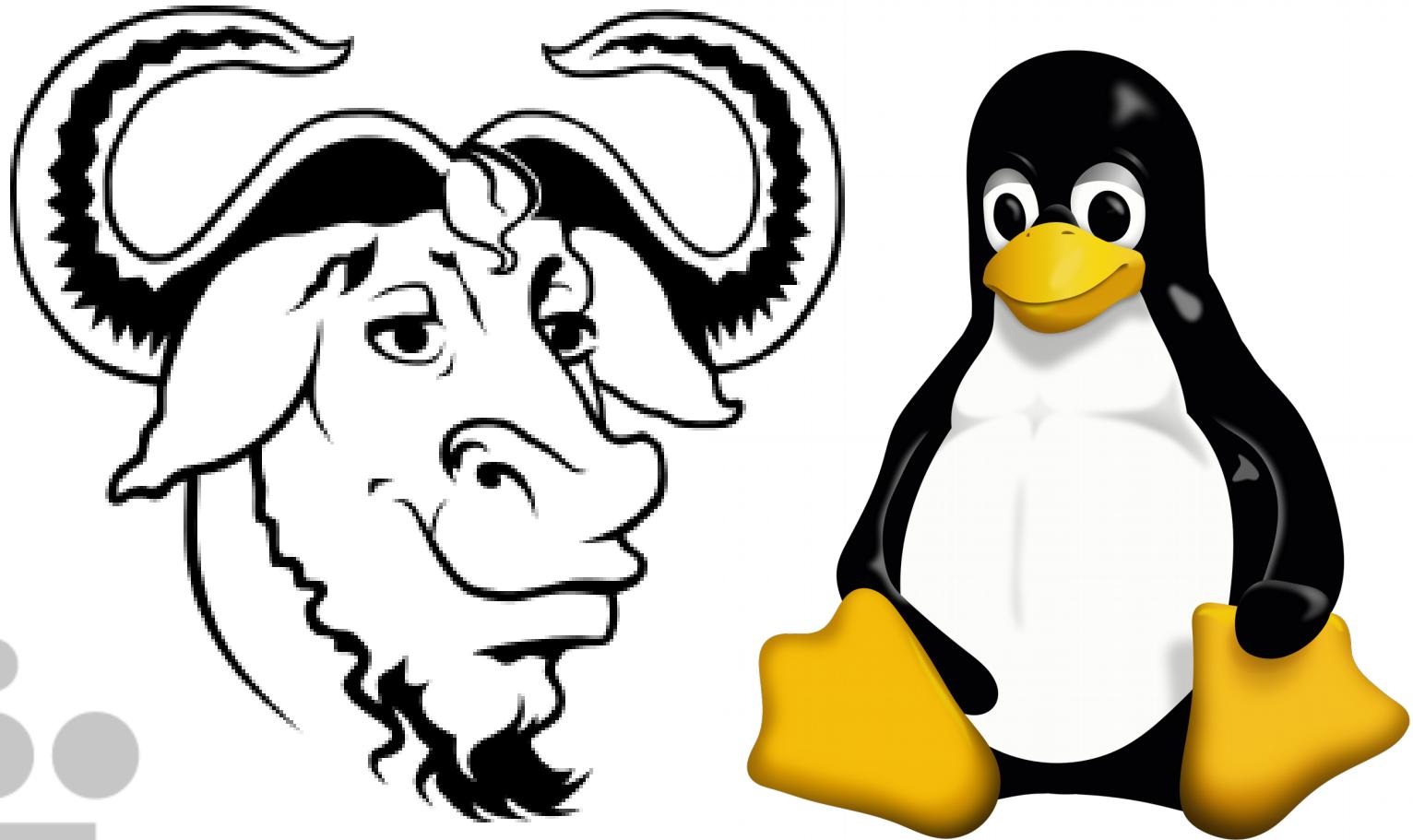
online

Niveles crecientes de complejidad (grado, máster, doctorado)



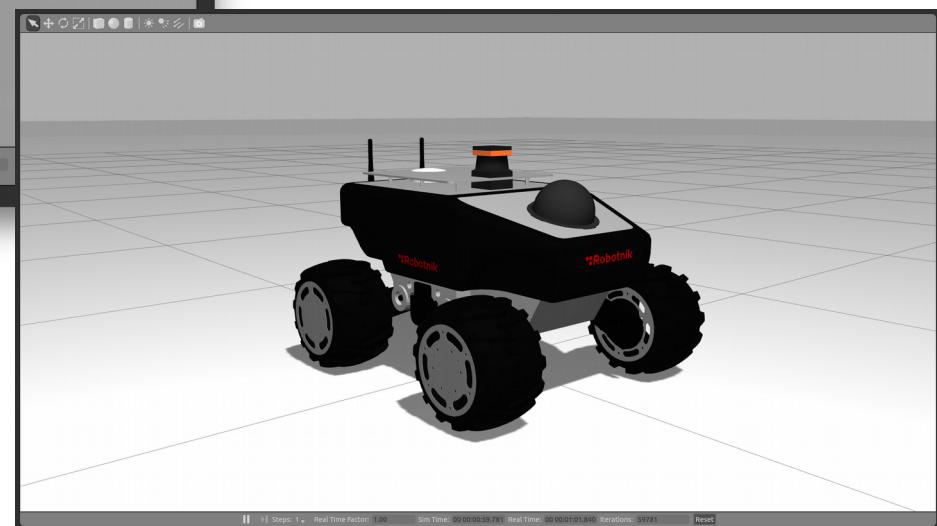
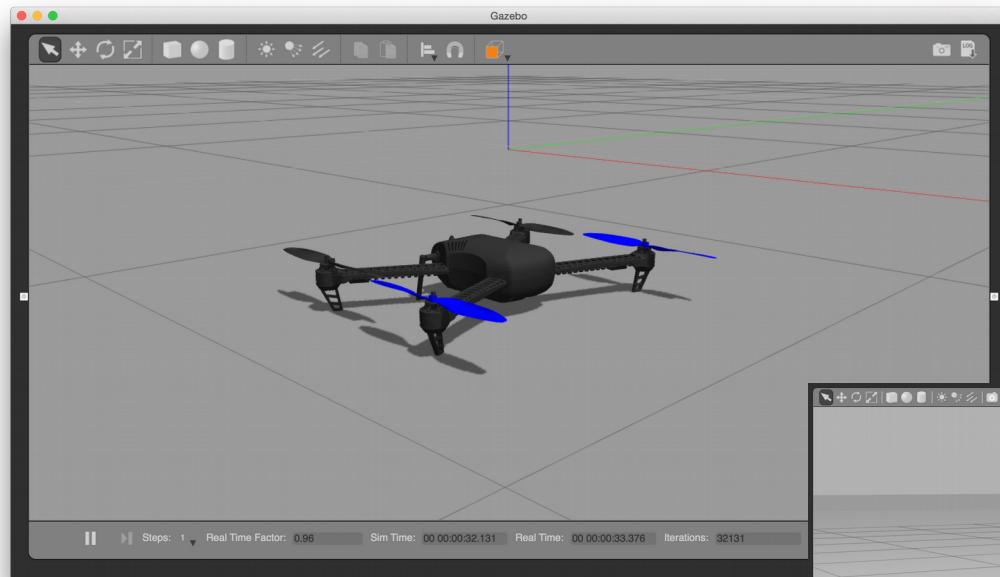
```
31 //OpenCV
32 #include <opencv2/core/core.hpp>
33 #include <opencv2/imgproc/imgproc.hpp>
34 #include <opencv2/highgui/highgui.hpp>
35
36
37 #include <string.h>
38 #include <iostream>
39 #include <cstdio>
40 #include <csignal>
41 #include <unistd.h>
42 #include <stdlib.h>
43 #include <list>
44
45 #include <zlib.h>
46 #include <logger/Logger.h>
47 #include <jderobotutil/interfaceHandlers/CameraHandler.h>
48 #include <jderobotutil/interfaceHandlers/CameraTask.h>
49 #include <ns/ns.h>
50
51 #include "easyiceconfig/EasyIce.h"
52
53 bool flag=false; /* boolean to keep a check on signal */
54
55 namespace cameraserver{
56
57 class CameraI: public jderobot::CameraHandler {
58 public:
59     CameraI(std::string propertyPrefix, Ice::CommunicatorPtr ic):jderobot::CameraHandler(propertyPrefix,ic){
60         //we use formats acording to colorspace
61         std::string fmtStr = prop->getPropertyWithDefault(prefix+"Format","YUV2");//default format YUY2
62         imageFmt = colorspaces::Image::Format::searchFormat(fmtStr);
63         if (!imageFmt)
64             throw "Format " + fmtStr + " unknown";
65
66         imageDescription->size = imageDescription->width * imageDescription->height * CV_ELEM_SIZE(imageFmt->cvType);
67         imageDescription->format = imageFmt->name;
68
69         // mirror image
70         mirror = prop->getPropertyAsIntWithDefault(prefix+"Mirror",0);
71
72         //fill pipeline cfg
73         uri = prop->getProperty(prefix+"Uri");
74         framerateN = prop->getPropertyAsIntWithDefault(prefix+"FramerateN",25);
75         framerateD = prop->getPropertyAsIntWithDefault(prefix+"FramerateD",1);
76
77         std::cout << "URI: " << uri << std::endl;
78
79         if(uri.size()>3)
80             capture.open(uri);
81         else
```

Software libre



ii online

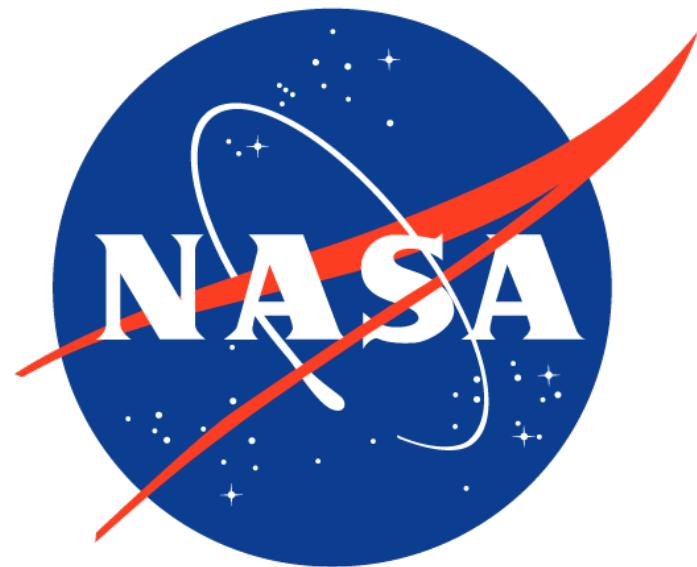
Simulación: prácticas heterogéneas



online



Lenguaje Python: Foco en el robot



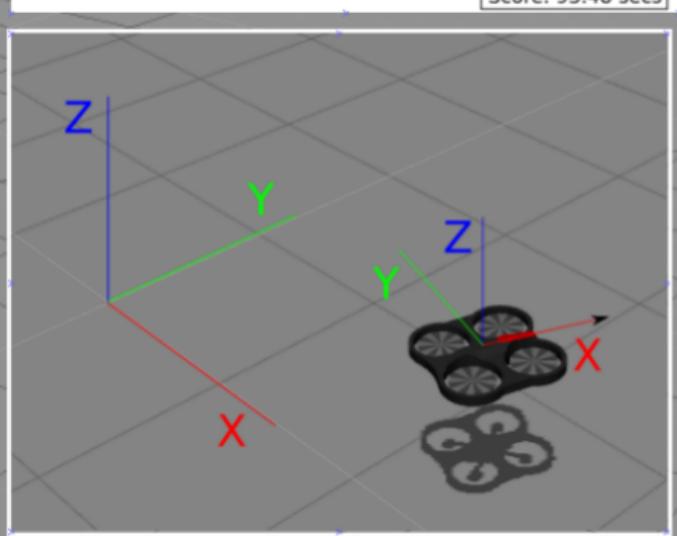
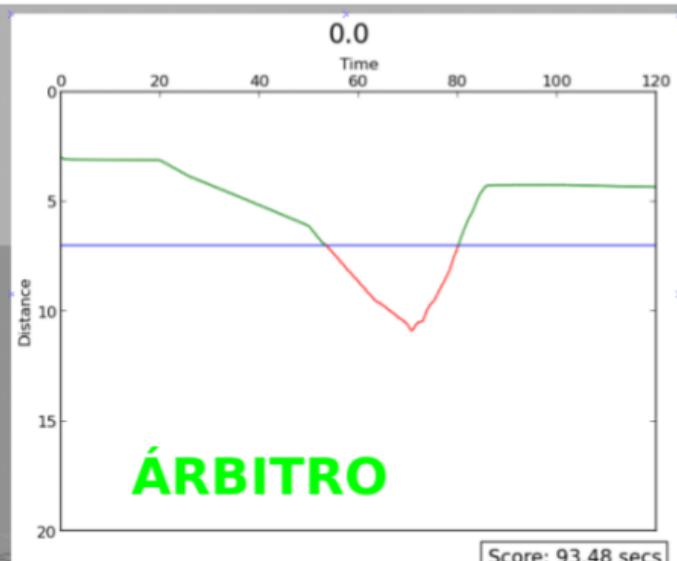
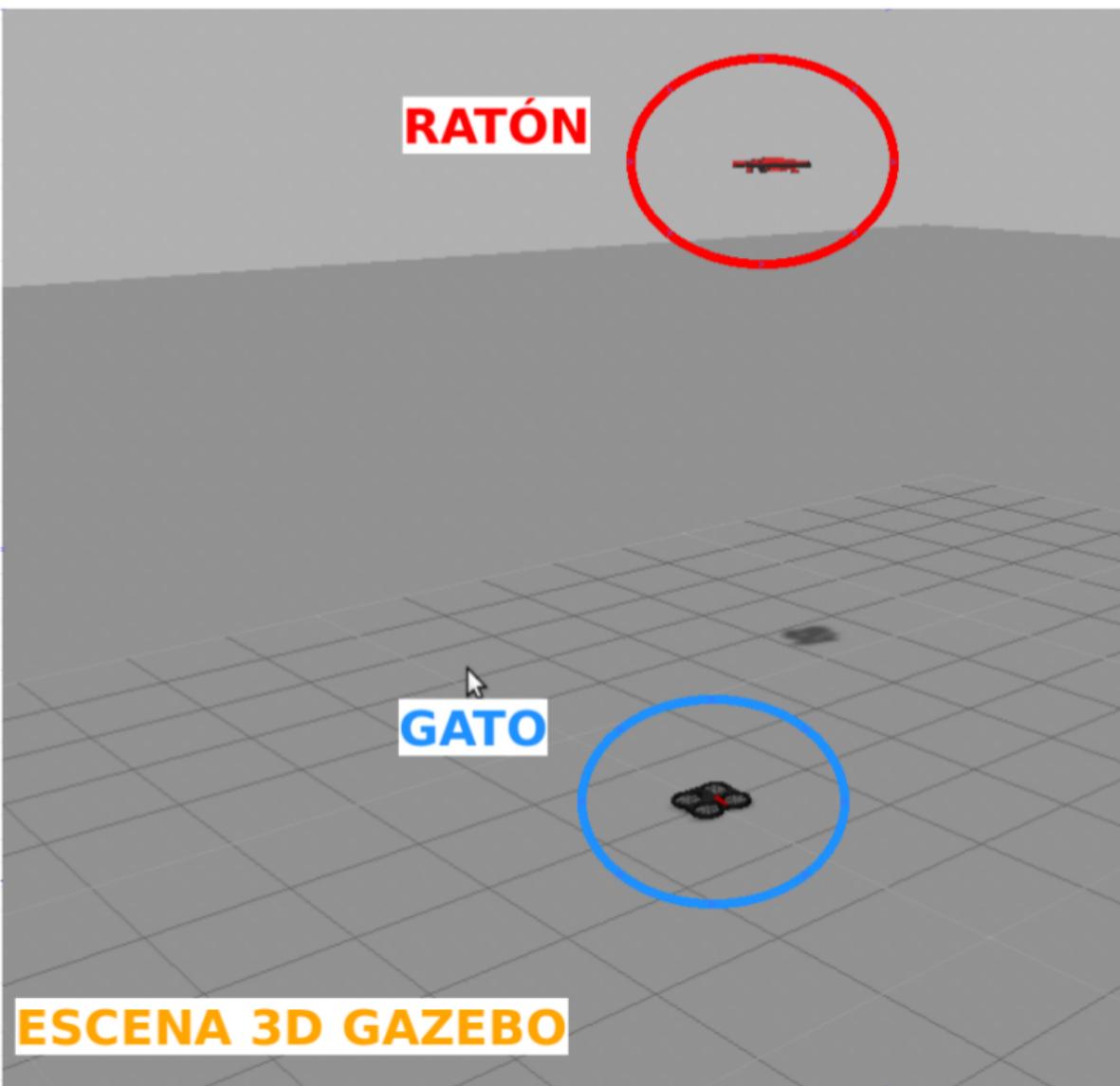
online

Gamificación: Prácticas como un juego



ííonline

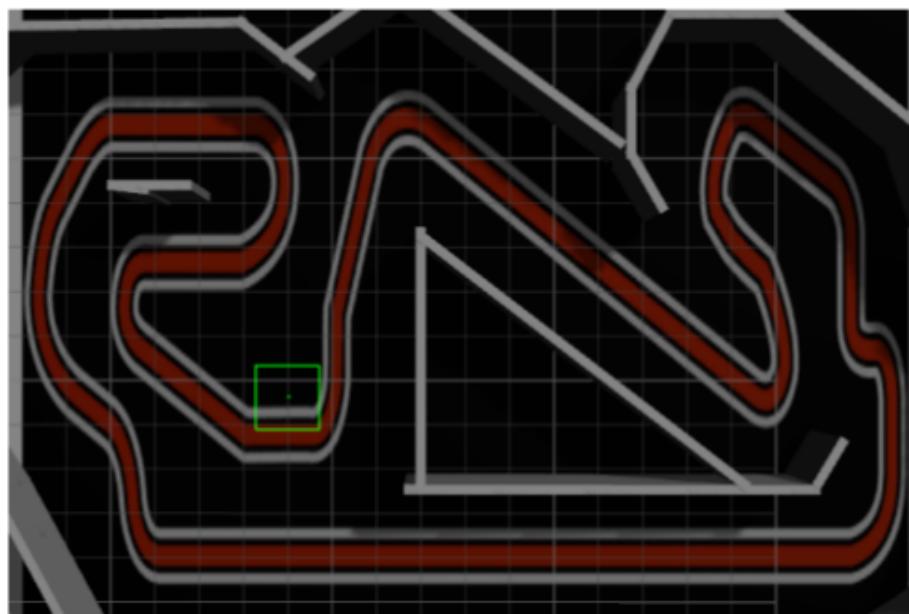
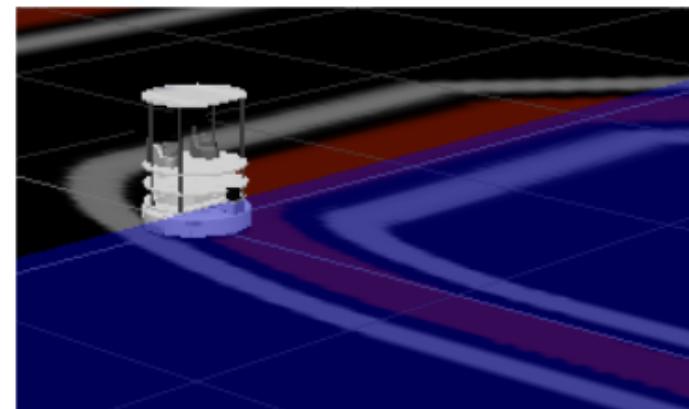
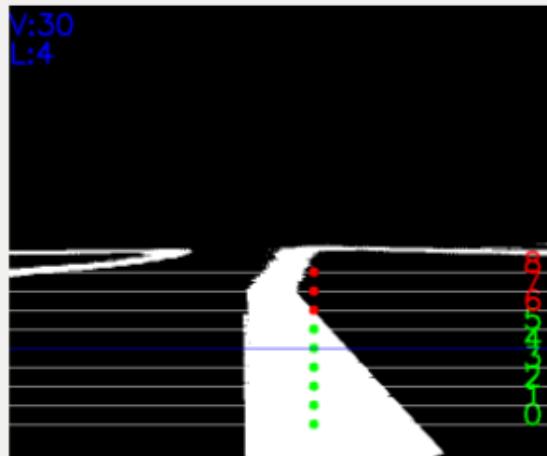
Práctica: Drones y persecución



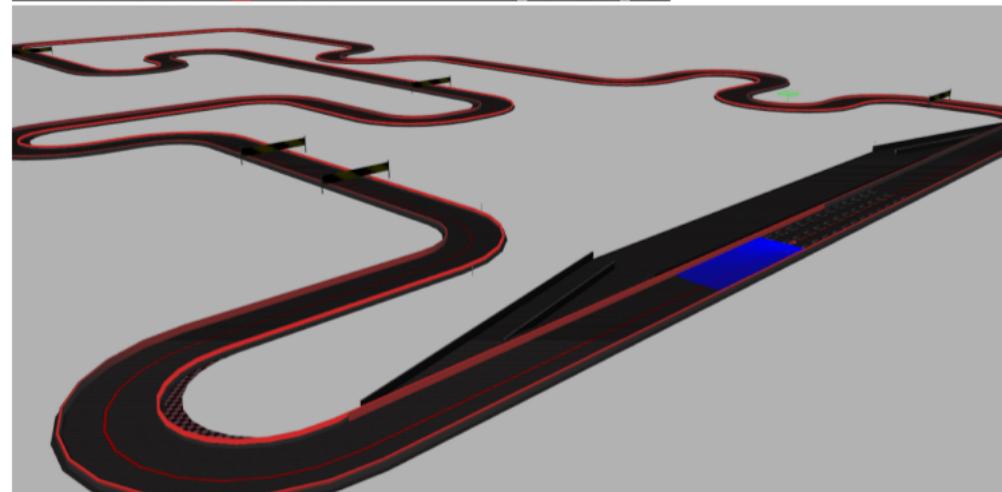
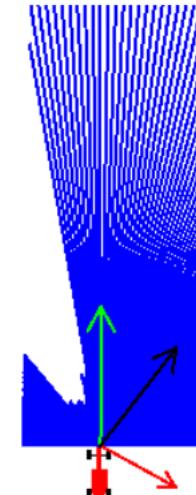
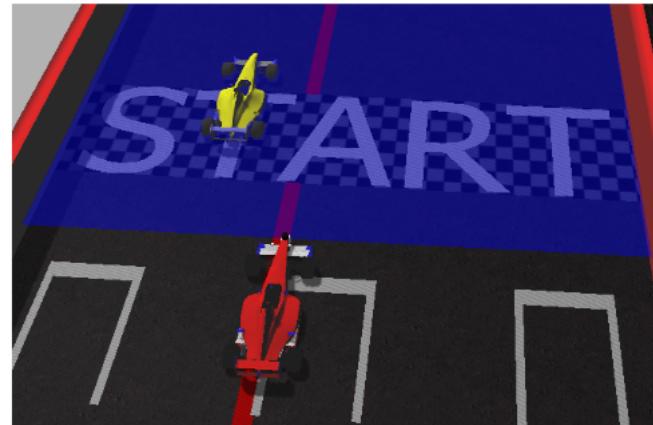
Práctica: Control Visual. Sigue líneas



Input

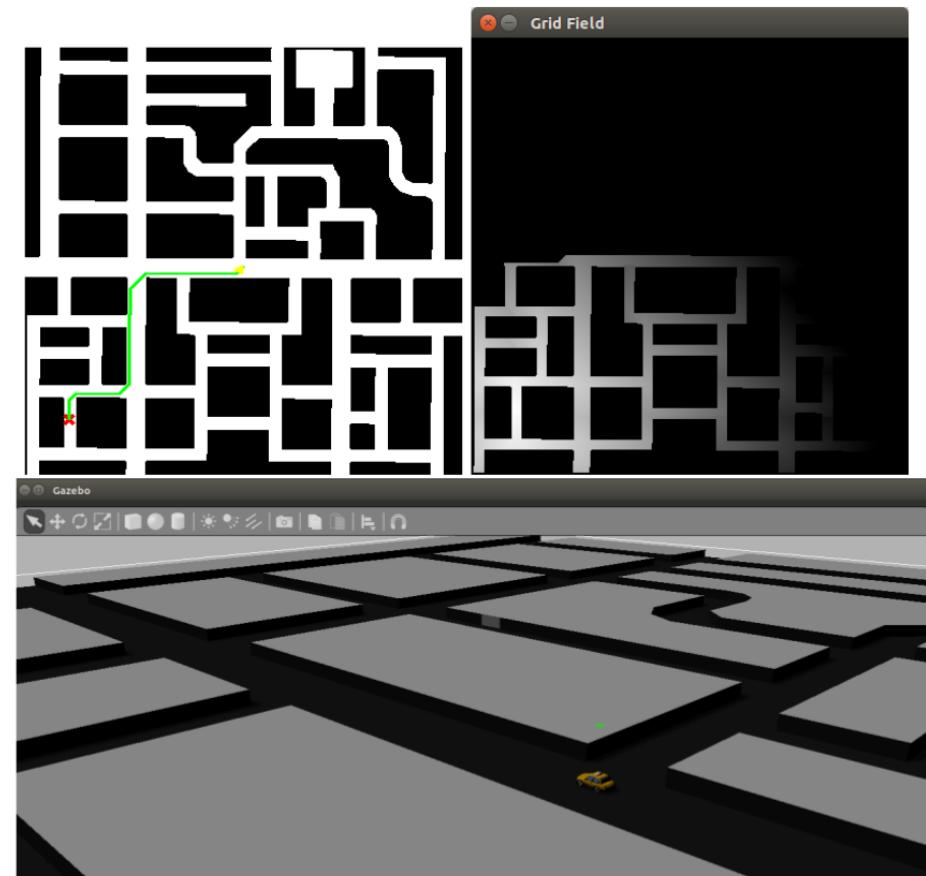


Práctica: Fórmula-1. Navegación local



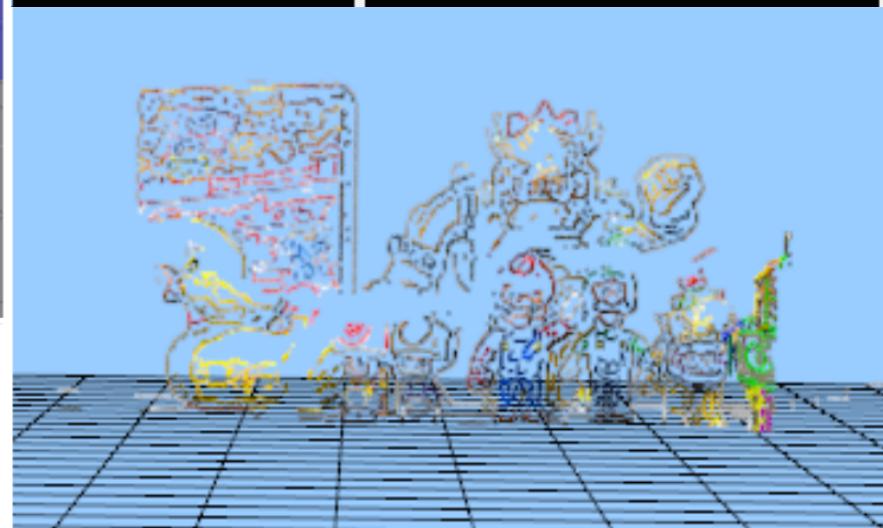
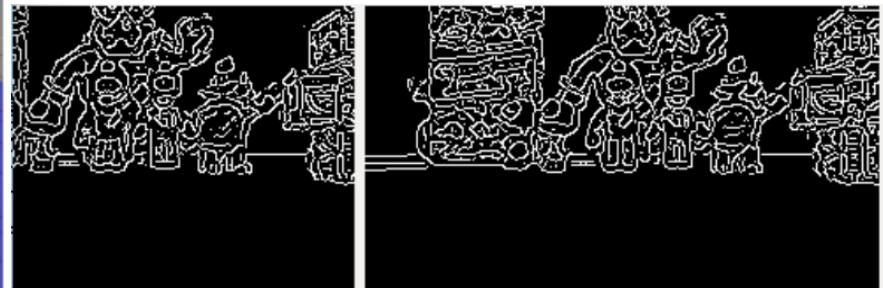
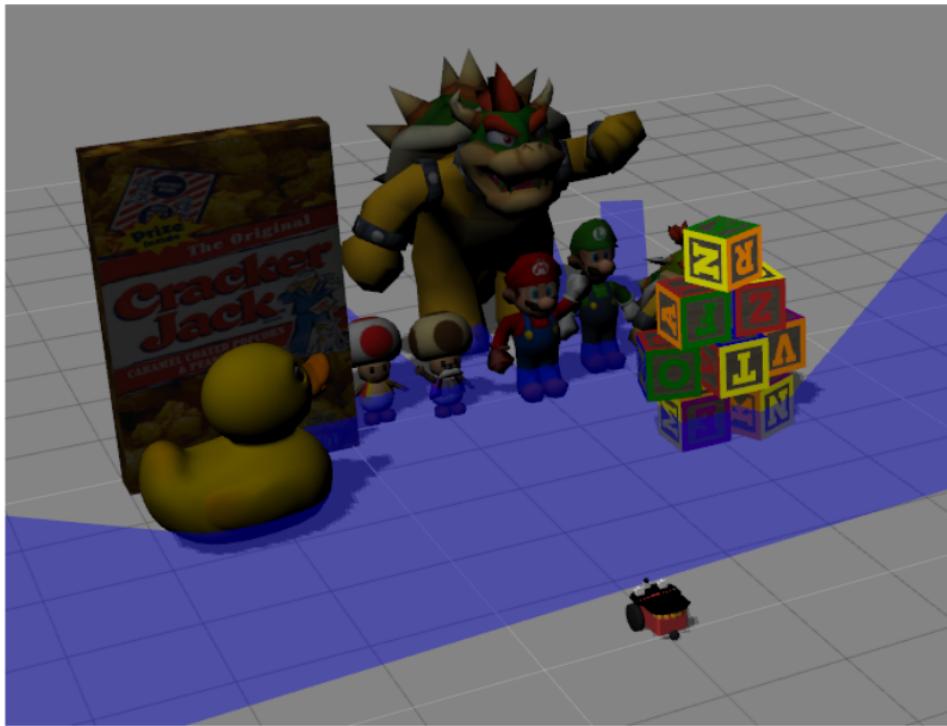
ii online

Práctica: TeleTaxi. Navegación global



online

Práctica: Reconstrucción 3D



online

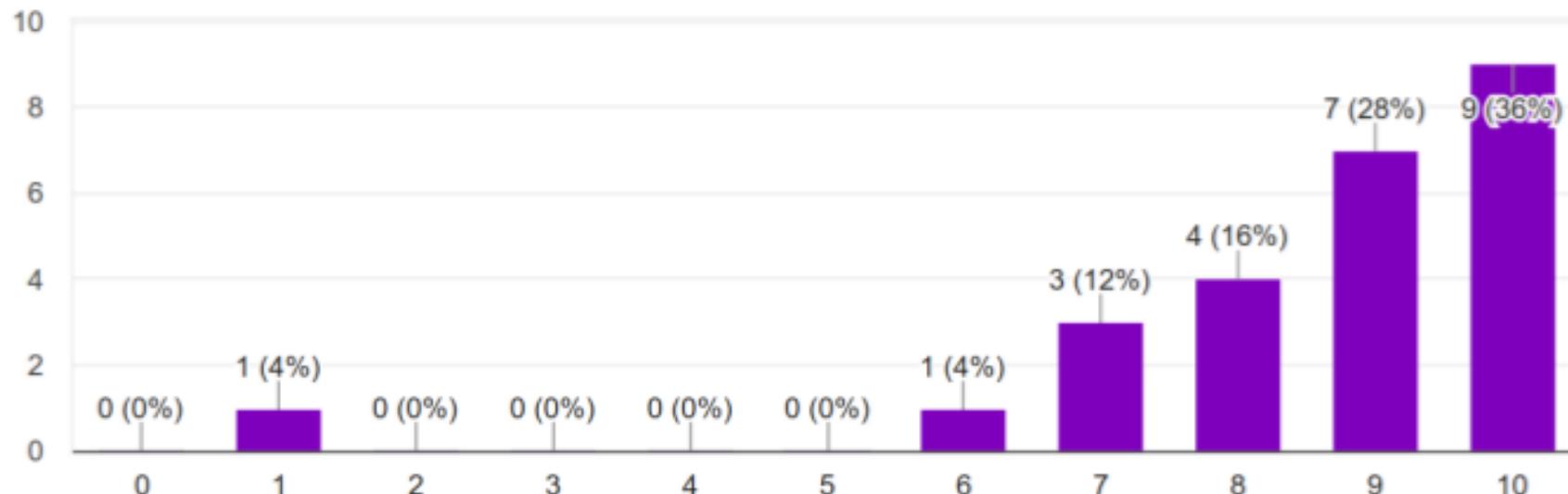
Resultados



¿Te han gustado las prácticas con JdeRobot?



25 responses



online



Futuro

ROS

Ü online

