

Sistema FIEB



PELO FUTURO DA INOVAÇÃO

CENTRO UNIVERSITÁRIO SENAI CIMATEC

Relatório dos desafios

Desafios: Webots, Turtlesim, CPP, Python e husky

Apresentada por: Marcella Giovanna Silva dos Santos

Orientador: Prof. Marco Reis, M.Eng.

novembro 2021

Marcella Giovanna Silva dos Santos

Desafios: Webots, Turtlesim, CPP, Python e husky

Salvador
Centro Universitário SENAI CIMATEC
2020

Resumo

Com o intuito de aperfeiçoar e aprender alguns conceitos aplicados na robótica, foi realizados cinco desafios além de todos os tutoriais de ros, os quais utilizam diferentes aplicações como: simulação de robôs, ROS, Webots e linguagens de programação (python e c++). Para que dessa forma seja possível a atuação no laboratório.

Palavras-chave: Conceitos, robótica, simulação, ROS, programação.

Abstract

In order to improve and learn some basic concepts in robotics, five challenges were carried out in addition to all the tutorials of ros, they use different applications such as: training simulation, ROS, Webots and programming languages(python and c ++). For what in this way it is possible to work in the laboratory

Keywords: Concepts, robotics, simulation, ROS, programming.

Sumário

1	Introdução	1
1.0.1	Webots	1
1.0.2	Turtlesim SETPOINT POSITION	1
1.0.3	Husky	1
1.0.4	CPP workbook	1
1.0.5	Python workbook1e2	2
1.1	Objetivos	2
1.1.1	Objetivos Específicos	2
1.1.2	Webots	2
1.1.3	Turtlesim SETPOINT POSITION	2
1.1.4	Husky	3
1.1.5	CPP workbook	3
1.1.6	Python workbook1e2	3
1.2	Justificativa	4
1.3	Organização do documento	4
2	Conceito do projeto	5
2.1	Webots	5
2.1.1	PIONEER	5
2.1.2	Sensor de distância	5
2.1.3	Sensor de Luminosidade	6
2.2	Turtlesim	6
2.2.1	tópicos ROS	6
2.3	Husky	6
2.4	CPP workbook	7
2.5	Python workbook1e2	7
3	Desenvolvimento do projeto	8
3.1	Webots	8
3.1.1	controle	8
3.2	Turtlesim	9
3.3	Husky	9
4	Resultados	10
4.1	Testes unitários	10
4.2	Integração do sistema	10
4.3	Testes integrados	11
5	Conclusão	12
5.1	Considerações finais	12
A	Diagramas mecânicos	13
B	Diagramas eletro-eletrônicos	14
C	Logbook	15

Referências**16**

Lista de Tabelas

Lista de Figuras

2.1	PIONEER	5
2.2	Representação do sensor de distância	6
2.3	Representação do sensor de luminosidade	6
3.1	Primeiro ciclo QFD	9

Introdução

1.0.1 Webots

Webots é um aplicativo de desktop de código aberto e multiplataforma usado para simular robôs. Ele fornece um ambiente de desenvolvimento completo para modelar, programar e simular robôs. Ele foi projetado para um uso profissional e é amplamente utilizado na indústria, educação e pesquisa. Cyberbotics Ltd. mantém Webots como seu produto principal continuamente desde 1998.

1.0.2 Turtlesim SETPOINT POSITION

Uma maneira simples de aprender o básico de ROS é usando o simulador turtlesim. Que consiste na simulação de uma janela grafica que mostra um robô com formato de tartaruga. Esta tartaruga pode ser movida por toda a janela utilizando comandos de ROS como `roscore`, `roslaunch`... ou utilizando o teclado/joystick.

1.0.3 Husky

O Husky é um veículo terrestre não tripulado (UGV) robusto, pronto para uso ao ar livre, adequado para pesquisas e aplicações de prototipagem rápida e oferece suporte total a ROS.

1.0.4 CPP workbook

O C++ é uma linguagem de programação de nível médio, baseada na linguagem C. Tem uma enorme variedade de códigos, pois além de seus códigos, pode contar com vários da linguagem C. Esta variedade possibilita a programação em alto e baixo níveis.

1.0.5 *Python workbook1e2*

Python é uma linguagem de programação de alto nível, ou seja, com sintaxe mais simplificada e próxima da linguagem humana, utilizada nas mais diversas aplicações, como desktop, web, servidores e ciência de dados.

1.1 *Objetivos*

- Webots: Desenvolver um sistema de navegação autônoma, de forma que o robô consiga chegar á região iluminada do mapa pré-definido, evitando todos os obstáculos do percurso em 2 minutos.
- Turtlesim: Fazer um controlador de posição para a tartaruga que da um ponto (X, Y) a tartaruga deve atingir o ponto com certo erro associado.
- Husky: Simular a navegação do robô Husky como está descrito no tutorial do ROS.
- CPP workbook: Demonstrar os conhecimentos adquiridos nos tutoriais de c++.
- Python workbook1e2: Fornecer uma oportunidade para que todos evoluam no conceito de codificação.

1.1.1 *Objetivos Específicos*

1.1.2 *Webots*

Os objetivos específicos deste desafio são:

- Fazer navegação do PIONEER;
- Utilizar o repositório <https://github.com/Brazilian-Institute-of-Robotics/desafiorobotica.git>;
- Utilizar sensor de luminosidade;
- Simular no webots;

1.1.3 *Turtlesim SETPOINT POSITION*

Os objetivos específicos deste desafio são:

- A posição do ponto de ajuste deve ser definida por meio de linha de comando;
- A escolha da linguagem de programação é Python;
- O código deve estar disponível em um repositório GitHub;
- O erro aceito é em torno de 0,1;

1.1.4 *Husky*

Os objetivos específicos deste desafio são:

- Fazer o tutorial Husky;
- Utilizar os pacotes do Husky;
- Simular a navegação nos 4 modos;
- Simular usando o gazebo;

1.1.5 *CPP workbook*

Os objetivos específicos deste desafio são:

- Fazer os desafios propostos;
- Utilizar C++ para a resolução;
- O código deve estar disponível em um repositório GitHub;

1.1.6 *Python workbook1e2*

Os objetivos específicos deste desafio são:

- Fazer os desafios propostos;
- Utilizar Python para a resolução;
- Cada exercício deve ser preparado na menor quantidade de linhas possível;
- Os comentários são necessários para você explicar cada etapa do código;

1.2 Justificativa

Demonstrar os conhecimentos adquiridos dos topicos de Webots,ROS,CPP e python durante o processo de todos os desafios, a fim de ter um ótimo desempenho nas atividades do Laboratório.

1.3 Organização do documento

Este documento apresenta 5 capítulos e está estruturado da seguinte forma:

- **Capítulo 1 - Introdução:** Contextualiza o âmbito, no qual os desafios estão inseridos. Apresenta, portanto, a definição dos desafios, objetivos e justificativas do mesmo, alem de como este relatório dos desafios está estruturado;
- **Capítulo 2 - Fundamentação Teórica:** As teorias usadas em cada um dos desafios;
- **Capítulo 3 - Materiais e Métodos:** De que forma os desafios foram feitos;
- **Capítulo 4 - Resultados:** Os resultados obtidos em cada um dos desafios propostos;
- **Capítulo 5 - Conclusão:** Apresenta as conclusões.

Conceito do projeto

2.1 *Webots*

O software Projeta facilmente simulações robóticas completas usando a biblioteca de ativos Webots que inclui robôs, sensores, atuadores, objetos e materiais.

2.1.1 *PIONEER*

Veículo terrestre não tripulado (VTNT) da empresa Adept MobileRobots. São equipados com sonares e encoders, possibilitando, por exemplo, o mapeamento de um ambiente desconhecido.

Figura 2.1: PIONEER

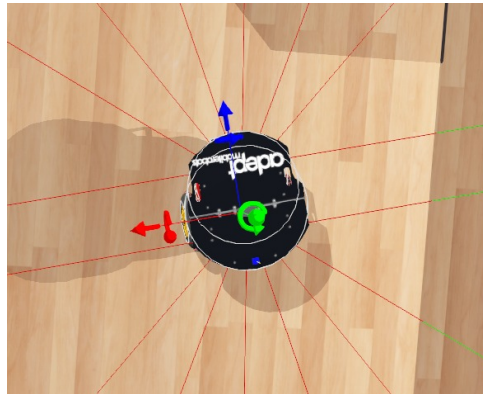


Fonte: robotica.ufv.br/laboratorios/.

2.1.2 *Sensor de distância*

O sensor de distância, é um tipo de sensor que mede a distância entre o robô e um objeto desejado. Abaixo pode ser visto 16 feixes que representam as posições dos sensores de distância do Pioneer no Webots.

Figura 2.2: Representação do sensor de distância

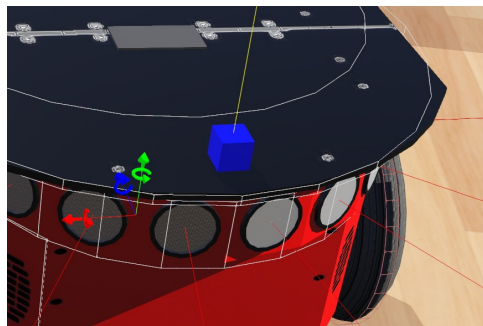


Fonte:Própria.

2.1.3 Sensor de Luminosidade

sensor é medir a intensidade de luz do ambiente ao seu redor, variando o estado de sua saída digital caso detectado um determinado nível de luminosidade. Abaixo pode ser visto um cubo amarelo que representa o sensor luminosidade adicionado no Pioneer.

Figura 2.3: Representação do sensor de luminosidade



Fonte:Própria.

2.2 Turtlesim

2.2.1 tópicos ROS

2.3 Husky

Além disso, o Walker deve realizar um desafio, que consiste em navegar de forma autônoma, se localizar por meio de tags e encontrar um determinado objeto.

2.4 *CPP workbook***2.5** *Python workbook1e2*

Desenvolvimento do projeto

Nesta seção será descrito o procedimento utilizado para construção de cada um dos desafios.

3.1 *Webots*

O robô utilizado foi o Pioneer o qual usa seus 16 sensores de distância pré-instalados para obter informações do mundo em todas as direções e um sensor adicional: o sensor de luminosidade que rastreia a irradiância local e envia um sinal para o robô quando ele lê mais de $750 \text{ W} / \text{m}^2$, o que significa que está perto o suficiente da luminária de chão para disparar o STOP.

3.1.1 *controle*

A navegação do robô pelo mapa é baseada em uma máquina de 4 estados que determina se ele deve se mover para frente, virar à esquerda, direita ou parar quando atingir seu objetivo final.

Assim, a divisão foi feita em quatro casos, são eles:

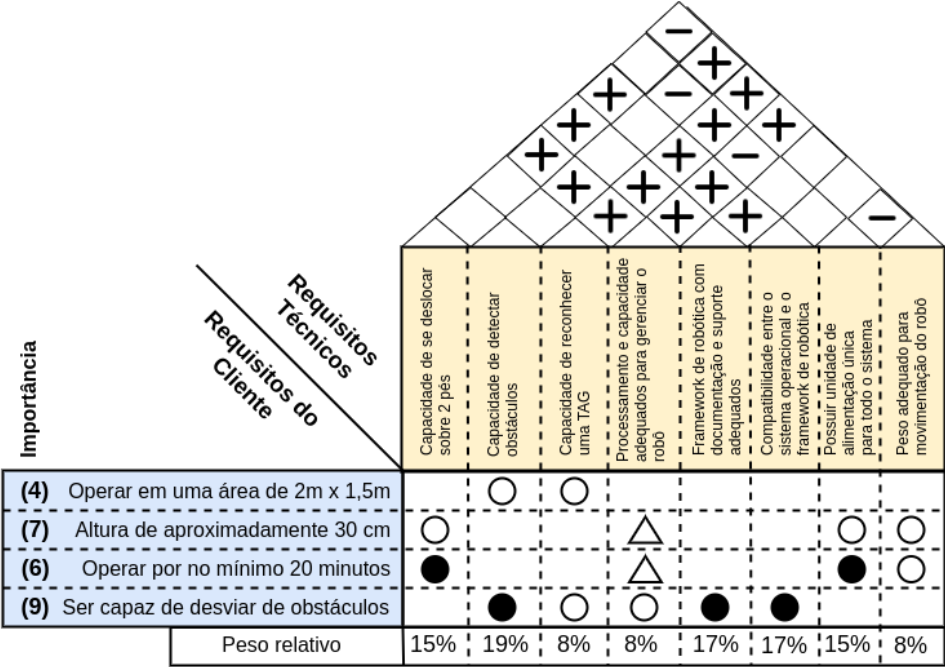
- FORWARD: Anda para frente e se houver algum obstáculo à frente ele começa a tomar a decisão de virar em qualquer direção para evitá-lo.
- ESQUERDA: Vire à esquerda até que a detecção de objetos não seja mais possível;
- DIREITA: Vire à direita até que a detecção de objetos não seja mais possível;
- STOP: Quando o sensor de luz detecta a quantidade de luminosidade configurada (neste caso $750 \text{ W} / \text{m}^2$), o robô deve parar.

3.2 Turtlesim

3.3 Husky

Quality Function Deployment é uma ferramenta de qualidade que auxilia na conversão das demandas do cliente em características de qualidade do produto. Dessa forma, no primeiro ciclo do QFD foram analisados os requisitos do cliente e os requisitos técnicos necessários, sinalizando os pontos mais importantes e as relações entre estes. O resultado foi exposto na 3.1

Figura 3.1: Primeiro ciclo QFD



Fonte: Autoria própria.

Através do QFD foi possível observar

Resultados

Importante sempre ter um parágrafo introdutório para explicar os resultados encontrados.

4.1 Testes unitários

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

4.2 Integração do sistema

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

4.3 *Testes integrados*

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Conclusão

Chegou a hora de apresentar o apanhado geral sobre o trabalho de pesquisa feito, no qual são sintetizadas uma série de reflexões sobre a metodologia usada, sobre os achados e resultados obtidos, sobre a confirmação ou rechaço da hipótese estabelecida e sobre outros aspectos da pesquisa que são importantes para validar o trabalho. Recomenda-se não citar outros autores, pois a conclusão é do pesquisador. Porém, caso necessário, convém citá-lo(s) nesta parte e não na seção seguinte chamada **Conclusões**.

5.1 Considerações finais

Brevemente comentada no texto acima, nesta seção o pesquisador (i.e. autor principal do trabalho científico) deve apresentar sua opinião com respeito à pesquisa e suas implicações. Descrever os impactos (i.e. tecnológicos, sociais, econômicos, culturais, ambientais, políticos, etc.) que a pesquisa causa. Não se recomenda citar outros autores.

Diagramas mecânicos

Diagramas eletro-eletrônicos

Logbook

Referências

Desafios: Webots, Turtlesim, CPP, Python e husky

Marcella Giovanna Silva dos Santos

Salvador, novembro 2021.