Product: PAUL
Team: Group 6

Abstract

In our first demonstration, we hope to successfully present a Webots Simulation of our robot climbing a straight vertical pole. We hope that this will demonstrate to the client the basic movement capabilities of our robot and give them a visualisation of how our robot will look.

1. Project plan update

1.1. Demonstration Goal

Our goal for this demonstration was to implement a Webots simulation of the pole climbing section of our robot which will be able to climb a straight pole with a consistent diameter. This simulation aims to demonstrate our robot's basic pole climbing movement.
Goal Outcome: **Achieved**

1.2. Milestones

The milestones each sub-team intended to complete for this demonstration were as follows:

- Software and Simulation Team:
  - 1.1 Simulate Climbing Part of the Robot
    - Success Criteria: the simulated robot moves smoothly at a designated and constant. speed up/down the pole.
    - Milestone Outcome: **Achieved**
- Front End/ Web Design Team:
  - 2.1 Skeleton Website
    - Success Criteria: website is structured and created.
    - Milestone Outcome: **Achieved**
- Hardware Team:
  - 3.1 Design a Case for the robot
    - Success Criteria: has a feasible design for the physical robot case and has discussed the manufacturing process with the technicians.
    - Milestone Outcome: **Achieved**
- Project Management Team:
  - 4.1 Prepare for Demo 1
    - Success Criteria: prepare the robot (tests conducted when needed) and materials for presentation.
    - Milestone Outcome: **Achieved** (When the video is submitted)
  - 4.2 Complete Demo 1 Report
    - Success Criteria: has LaTeX report finished and peer reviewed by all members.
    - Milestone Outcome: **Achieved** (When this report is submitted)

Our group is on track and we have achieved all our goals and milestones for this demonstration and currently, we are still on track to meet the milestones required for the second demonstration.

1.3. summary

To organise our work towards the goals we started the week off by having a meeting of the entire group. In this meeting, we discussed our mentor's feedback and decided to change a couple of things. Firstly, for better coordination and accountability, we decided that we should have 'leaders' or project managers for each sub-team. These leaders would be responsible for coordinating the work and tracking the progress of the sub-team. The tasks and milestone for each sub-team were added to our Trello board.

The team leaders are as follows:

- Project Management Team - Daniel
- Front-End/Web Design Team - Gagan
- Software and Simulation Team - Stirling
- Hardware Team - Pablo

Other than this, each sub-teams divided their tasks, scheduled meetings over the semester and updated the responsibilities of each member on the Trello board. This allowed us to assign members to individual tasks and provided a transparent way to see who was doing what (See Table 1) .

Finally, a new addition to the Trello sub-board was added. To ensure correctness and efficiency of our code, we decided to introduce a Quality Assurance card following T. Slawiz's article. (Slawiz, 2020)

On Friday, we had another entire group meeting where we discussed the progress each sub-team had made and checked we were on track for all our milestones. We reviewed the tasks and milestones using Trello. The members of the Project Management team then worked on a report and created a video to demonstrate the current implementation of our robot.

1.4. Testing and Code Integration

As explained in our project plan, we used Git and GitHub for code integration across both the Front-End and the Software and Simulations team, we followed the branch and merge strategy and fixed any code conflicts through code

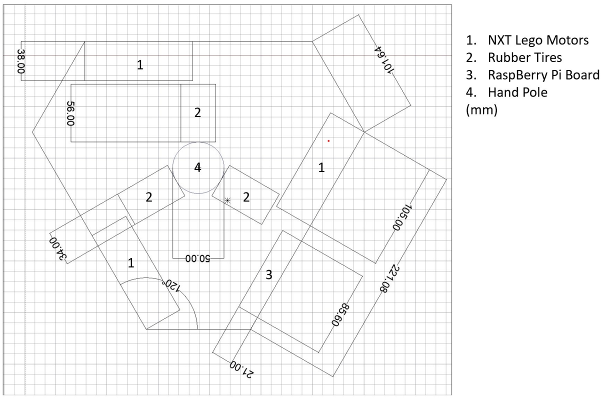

Figure 1. 2D Plans for the Hardware Implementation

reviews and pull requests as discussed by (Dwarkani, 2018). As the tasks were completed, we changed Trello cards to "Testing" stage and then to "Completed". A summary of the tasks can be found in table 1 as before. We followed the Agile protocol we learnt in Bloomberg's workshop and had lean stand-ups with the sub-teams and one integration meeting with everyone else.

2. Technical details

2.1. Hardware

We are currently working with the technicians to build our hardware implementation but in this section, we will explain what our current plan is.

For our first prototype, we aim to offload the robots' weight as much as it is possible, having a case with motors and wheels connected to the power supply with long cables, hence, avoiding unnecessary weight, focusing on lifting through a vertical pole a lightweight chassis.

The starting robot would be constructed from a LEGO case, allowing for an easier re-configuration should any changes be necessary for the future (such as adding thin plastic plates pegged directly to the LEGO or a 3D printed casing). For the motors the LEGO NXT will be used, having a rotation speed of 177rpm and a 16.7 N/cm(1.7 kg/cm) torque, should work fine with lifting the lightweight casing. Using a set of 8 AA batteries, with a capacity of 2700mAH each, should be sufficient for the first task.

If the lightweight robot performs as expected, we would move to a more sophisticated configuration. Adding an accelerometer to get readings on the speed, range sensor to know when the robot has finished cleaning the whole pole and gyroscope to be aware of the orientation of the wheels, among others.

The technical drawing (see Figure 1) with measures is to be sent to technicians to start the manufacturing of the chassis. A meeting with the technicians is organised for Monday at 10:30 am, to set up the Raspberry Pi and get readings from the motors.

2.2. Software

The software developed this week was entirely using Webots to achieve a simple simulation of the robot. We defined the success criteria for completing milestone 3.1 as "simulated robot moves smoothly at a designated and constant speed up/down the pole." We tested this by measuring the amount of time that the robot took to ascend/descend the pole with different speed parameters.

To do this, we had to overcome several challenges. This report will discuss these in about their tasks.

1. **Simulate Environment**
   We experienced no difficulties in creating poles although in the simulation we decided to implement a physics plugin - this had to be done in C/C++ - which would stabilise the poll by creating a joint between it and the world. Different materials were defined for the wheels and the pole respectively, so that friction could be defined between them, allowing the robot to climb more realistically we used (Shiffman, 2012) to aid our understanding of simulating physics. Generally, objects are implemented as shapes with bounding boxes, with physics objects applied.

2. **Simulate simple robot case**
   Implemented using a group object so that compound shapes can be defined and allowing easier implementation of physics. This layout can be seen in Figure 2.

3. **Simulate robot wheels**
   Implemented as Hinge Joints using rotational motors. Initially, we experienced difficulty in understanding axis and anchor parameters although this was later resolved through reading documentation.

4. **Write controller to simulate climbing**
   Initially, we implemented a two-wheeled robot which differed from the design set out by the hardware team because we perceived this to be a simpler goal with fewer complexities. Using this robot we successfully simulated pole ascent and descent.

5. **Implement altering speed**
   Currently implemented in the controller, can be edited although in future we hope to make this possible using a web-based UI.

6. **Implementing additional rotated wheels**
   After successfully implementing our two-wheeled robot, through fully understanding the Hinge Joint Parameters class and other sub classes of Hinge Joint we were able to complete our three wheeled robot - as specified by the hardware team - implemented will all the same functionality. Namely by editing the axis and anchor fields to allow for proper rotation of wheels.

7. **Implementing circular pole**
   Implemented similarly to square pole used, utilising the same physics plugin.

Milestone	Task/Aspect	Member
1.1	Simulate environment	Stirling
	Simulate robot case	Stirling
	Simulate robot wheels	Stirling, Jim
	Write controller to simulate climbing	Stirling
	Implementing altering speeds	Stirling
	Implementing additional rotated wheels	Yasmin
	Implement circular pole in addition to square one.	Yasmin
	Implement distance sensors	Jim
	Editing controller to work with sensors	Jim, Yasmin
2.1	Discuss website features with the entire group	Gagan
	Vote on a starter-skeleton template	Zhiqi, Jim and Gagan
	Choose Colour theme for the marketing website	Zhiqi
	Plan Website pages	Gagan
	Implement individual web pages	Jim and Zhiqi
	Add team members	Gagan
	User Testing of each web page	Zhiqi, Jim and Gagan
3.1	Produce a technical drawing of the robot case and components	Pablo
	Discuss manufacturing of case with technicians	Pablo, Daniel
4.1	Prepare for the Q&A on Wednesday	Daniel, Stirling
4.2	Project Plan Update Section	Daniel, Gagan
	Demo Report Software Section	Stirling
	Demo Report Hardware and Budget Sections	Pablo
	Demo Report User Interfaces Sections	Gagan
	Demo Report Evaluation Section	Stirling
	Create intro section and project management section of report video	Daniel
	Create a Software section for report video	Stirling
	Create a Hardware section for report video	Pablo
	Edit report video sections together	Daniel
	Check and Submit Report	Daniel, Gagan

Table 1. Contributions of each member towards tasks

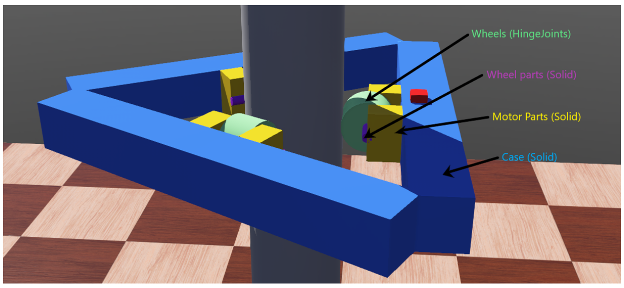

Figure 2. Layout of the Webots Simulation Implementation Parts

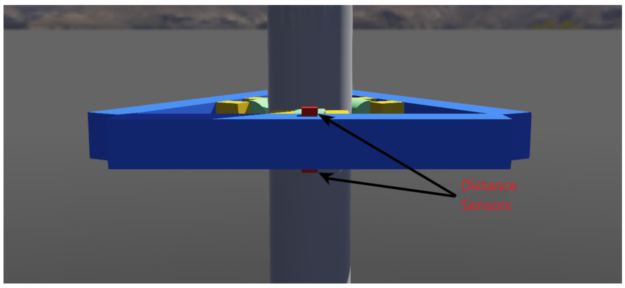

Figure 3. Layout of the Webots Simulation Implementation Distance Sensors

8. **Implementing distance sensors**
Two distance sensors were added using Shape objects and the Distance Sensor class, one on the top and one on the bottom to detect collisions. The positioning of these sensors can be seen in Figure 3.

9. **Editing controller to work with sensors**
Using the output from the two distance sensors we were able to enable the controller to recognise when a collision was about to occur and either stop or move in the opposite direction. (Sensitivity in this regard can be adjusted)

2.3. User Interface

So far the software and simulations worked with understanding how Webots works and getting the simulation ready for the demo, the front-end team focused on this component exclusively from the marketing website to the interactive web app that the controller (user) will be using to operate the device.

With perspective to research and reading, our front team read different components to design our systems better. Zhiqi studied about Colour Palettes (ShutterStock) and designing accessible user interfaces. (Hausler) Gagan read parts of Design of Everyday things (Norman, 2002) and learnt some JavaScript that could be helpful later on. While some of these relates to our current goals, we hope this will be useful later on in the course when we build the web app with React and all of this come together.

2.4. Marketing Website

We achieved all our goals for our marketing website, we used a open-source template to get us a flexible bootstrap to begin with. [2] Then, Gagan, Zhiqi, and Jim collaboratively worked on it to match our requirements. We deployed our website through GitHub Pages statically directly from the team's repository team's repository. The team followed the branch and merge strategy to work productively at our own pace with a fixed deadline for each events set by the team leader / project manager.

Finally, the front-end team deployed the blogs feature. The project managers will use this platform to give a weekly update informally to any of our users. However, this blog is also meant to be reflective in nature and focus on how the team can improve, while pondering about the challenges faced in the previous week.

3. Evaluation

Our evaluation was conducted on our Webots Simulation of the robot moving up and down the pole.

1. **Velocity Testing** + The pole was 2m high in the following tests, with a radius of 0.03m. Each test was conducted 5 times and each test number is an average of those five tests, so in total 25 tests were conducted of different speeds.

Test	Velocity(rad/s)	Time Taken (Seconds)	Success
1	1	123.20	√
2	3	38.65	√
3	5	23.50	√
4	7	16.57	√
5	9	13.12	√

Table 2. Velocity test

We were satisfied with these results as due to our intention to use UVC with the robot, it is very unlikely we would need to move this quickly.

2. **Height Testing**

Further tests were also conducted ensuring viability on poles of different lengths. All of these were conducted with a velocity of 5 (rad/s). Each test was run five times and the results did not differ between tests on the same length. The results were as follows, with success being defined as reaching the top of the pole after starting from the bottom.

Test	Height(m)	Success
1	1	√
2	5	√
3	9	√
4	10	×
5	20	×

Table 3. Height test

We noticed that at around 10m the robot would slip and fall down the pole, we continue to investigate this issue and plan to have it resolved by the next demonstration to support our alternative use cases, we currently believe it may be down to the robot's case and wheels not being entirely the correct size or rotation.

4. Budget

At the moment our expenditure amounts to £0. In the Hardware team, the initial set up of the prototype is built with components which are issued for free to all teams, mostly Lego. We predict that for the second demonstration we will have to order a custom casing, which will be issued to the technicians, as well as UVC light emitters, from uk.rs-online.com suppliers.

5. Video

Our video is available on the University SharePoint using these links: SharePoint Link and through a back up Youtube link.

References

Dwarkani, B. Pull requests and gitflow, 2018. URL https://blog.axosoft.com/pull-requests-gitflow/.

Hausler, Jesse. 7 things every designer needs to know about accessibility. URL https://medium.com/salesforce-ux/7-things-every-designer-needs-to-know-about-accessibility-64f105f08

Norman, Donald A. The design of everyday things. 2002. URL https://www.goodreads.com/book/show/840.The_Design_of_Everyday_Things.

Shiffman, Daniel. *The Nature of Code*. 2012.

ShutterStock. Coolors - colours palette. URL https://coolors.co/003049-6b2c39-d62828-eba141-ffcb3c-eae2b7.

Slawiz, Tadeudz. Why quality assurance is important, 2020. URL http://tinyurl.com/sdp-6-qa.