

# Spyingball

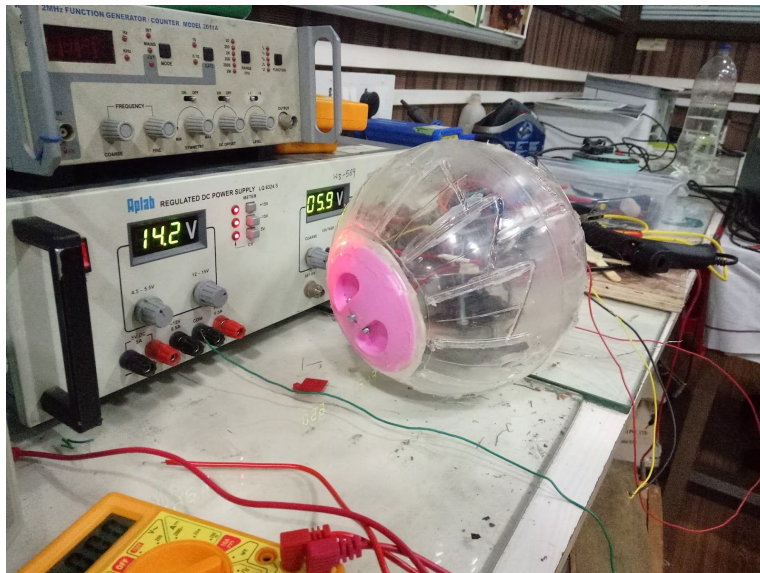
Team: GUESS WHO?

- Mrinal Dharmik
- Rohith Thakurwar
- Arvind Singh Arya

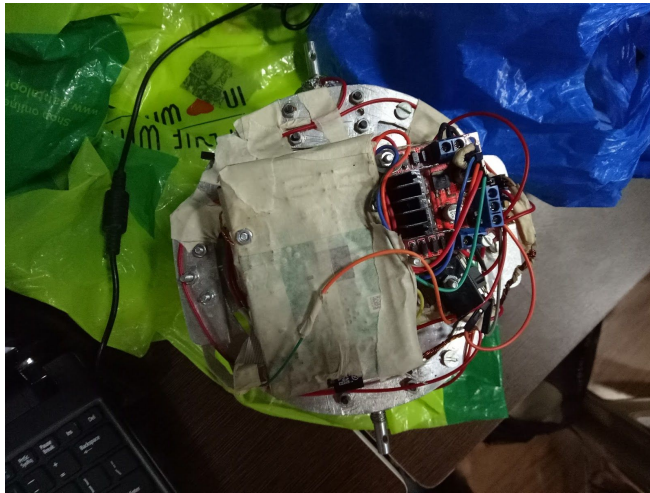
# Description:

The project is made using a hamster ball with Raspberry Pi 3. Rolling motion is produced by two 100 RPM motors which are on the main rolling axis. Sideways rolling is produced by another pair of 60 RPM motors which tilt the main axis by shifting the center of mass. The inner framework is made of aluminium and iron(to decrease the height of center of mass). The bot is further stabilised by two MPU6050 sensors on each axis. Raspberry Pi camera module is used for video streaming. The bot is controlled on the server side using php and javascript. The video stream can be seen on the control website itself. The bot has provisions for night vision spying as well.

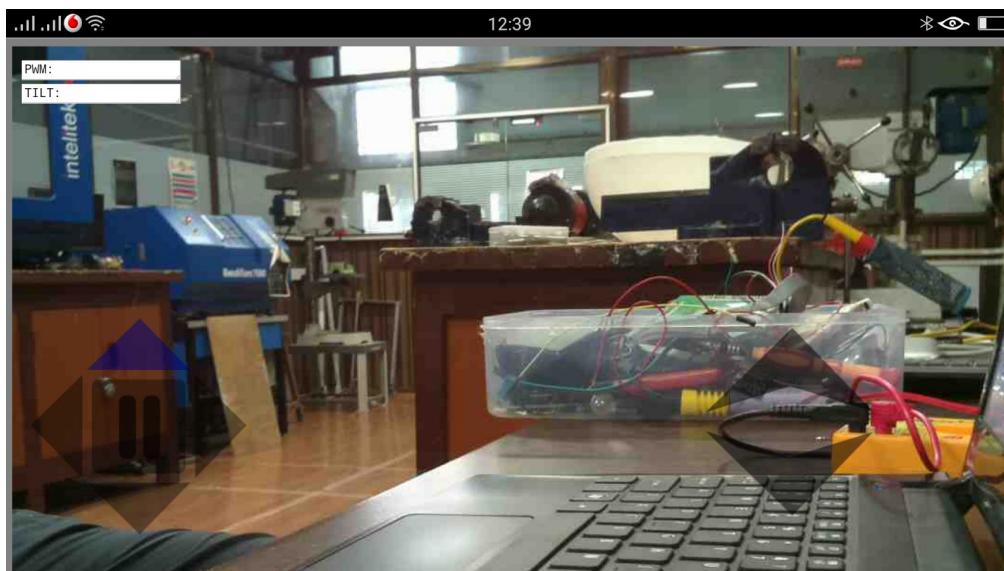
## Charging:

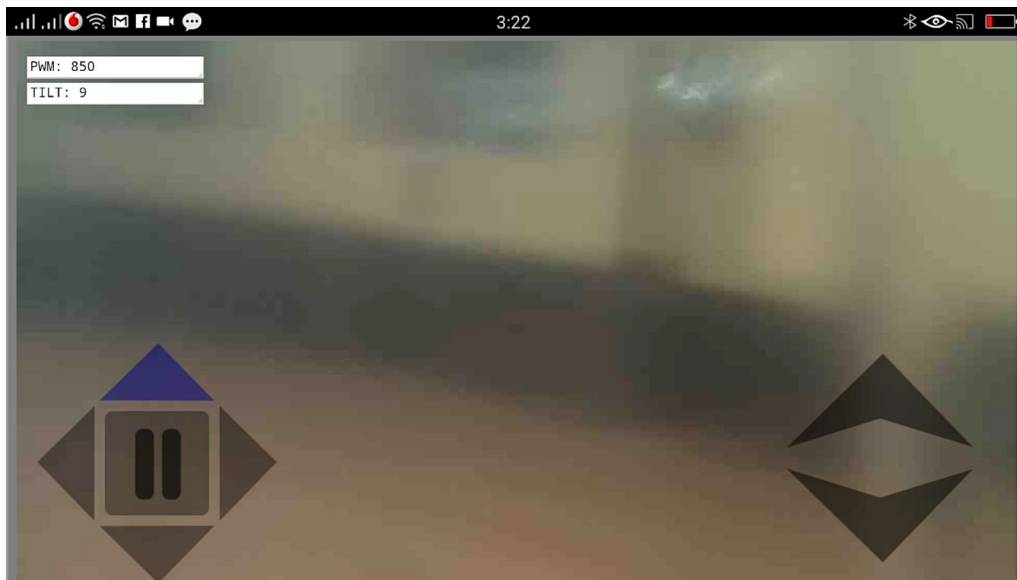
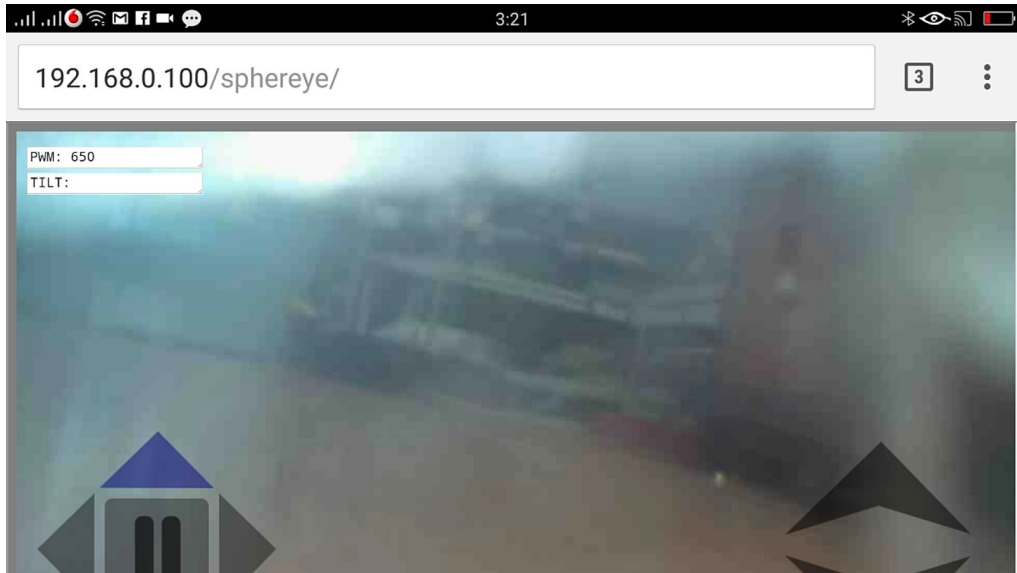


**Framework:**



**Web Controller(with streaming):** <https://github.com/Mrinaldo/Spyingball>





# Equipments:

- Hamster Ball
- Raspberry Pi 3
- Two MPU6050 sensors
- L298N motor driver
- 12V Li-ion Battery
- 2 100RPM motors
- 2 60RPM motors
- 2 to 3 sheets of aluminium
- 1 sheet of iron for extra mass
- 16 gb Class 10 SD Card
- Raspberry Pi NoIR Camera
- Jumper wires

# Timeline:

- From 25th May to 1st July: We researched possible mechanisms for ball movement and got ourselves familiar with raspberry pi. We tinkered with MPU6050 and got the accelerometer and gyroscope readings.
- From 1st July to 10th July: We decided upon the center of mass shifting mechanism and bought a hamster ball from a local pet store. We started building the inner framework of the ball and decided where we will place the raspberry pi, motor driver, battery, camera and extra weight.
- From 11th July to 20th July: We tested and debugged the mechanical part of the bot . Python code for motion was ready. We streamed the camera on vlc but there was a lot of time-lag. So we had to work on that during the next phase we had to concentrate on that.
- From 21st July to 30th July: We ditched the python and vlc combo and made a website for controlling the bot using php and javascript from scratch and embed the stream on that it self. We installed MPU6050 sensors and started stabilizing the bot with their help.

## Drawbacks:

- Time-lag in streaming.
- Code for the sensor control is not up to the mark will have to improve upon that.
- A better grip for smoother control.

## Possible Advancements:

- We will try and run it on water after sealing the ball completely.
- We can buy a better, high quality camera module for better video quality.
- Adding a microphone for better spying and surveillance.
- Look for a faster streaming method.

# Testing Videos:

- Test: <https://drive.google.com/open?id=0B--fCrocDAFUQ1k5RGFPY05vQVE>
- Test2: <https://drive.google.com/open?id=0B--fCrocDAFUdkRwQnkyMFIkcE0>
- Test3: <https://drive.google.com/open?id=0B--fCrocDAFUQ0lmdFJ4X2VkZ3M>
- Test4: <https://drive.google.com/open?id=0By7hvcUFa n9SNTThDMFpoaF9mU0U>
- Test5: <https://drive.google.com/open?id=0By7hvcUFa n9SY1BoQkRCWnhiLXM>
- Test6: <https://drive.google.com/open?id=0By7hvcUFa n9SRTdHR2FOWW5MU0k>
- Test7: <https://drive.google.com/open?id=0By7hvcUFa n9SU2ZjNVhwdmZLZ2M>
- Test8: <https://drive.google.com/open?id=0By7hvcUFa n9Sb1VEQ1dSeEsyZ1E>
- Test9: <https://drive.google.com/open?id=0B--fCrocDAFUcGtiQTNFNmInQXM>
- Test10: <https://drive.google.com/open?id=0B--fCrocDAFUUY3duc mNxdIR5Q3c>
- Screen-recording: <https://drive.google.com/open?id=0B--fCrocDAFUV2liQnNDY3ZpNEk>
- Final Video: <https://youtu.be/iZC5hDTvJ44>



# Important Links:

- Mechanisms: <https://www.youtube.com/watch?v=18S78rrMkr4>
- webcontrols: <http://www.instructables.com/id/Simple-and-intuitive-web-interface-for-your-Raspbe/step6/Conclusion-and-ideas-of-improvements/>
- WiringPi: <https://projects.drogon.net/raspberry-pi/wiringpi/download-and-install/>
- <http://raspberrypihq.com/how-to-turn-a-raspberry-pi-into-a-wifi-router/>
- Sensors: <http://www.instructables.com/id/Reading-I2C-Inputs-in-Raspberry-Pi-using-C/>