



COLEGIUL NAȚIONAL "GRIGORE MOISIL"

B-dul Timișoara nr. 33, Sector 6, București

Tel: 021 413 26 96; 021 413 26 47. Fax: 021 440 10 06

Website: <http://www.moisil.ro>

ROMÂNIA
MINISTERUL
EDUCAȚIEI

OLIMPIADA NAȚIONALĂ DE CREATIVITATE ȘTIINȚIFICĂ 2024

Secțiunea ȘTIINȚE APLICATE

Elevi:

Danilov Alexandru Cristian

Grigore Mihnea – Andrei

Țirulnic Serghei

Profesori coordonatori:

prof. Garabet Mihaela

Nr. Crt.	Unitate școlară/ Localitate/ Județ	Denumire proiect	Categorie	Secțiune	Membri echipă				Coordonator proiect			
					Nume prenume	Clasa	Unitatea școlară	Email	Nume prenume	Entitate reprezentată	Telefon	Email
1.	Colegiul Național „Grigore Moisil”, București, Sector 6	ATREX	Seniori	Științe Aplicate	Grigore Mihnea - Andrei	XII	Colegiul Național „Grigore Moisil”	amihnea2@gmail.com	Garabet Mihaela	Colegiul Național „Grigore Moisil”, București	0723084164	mihaela.garabet@gmail.com
					Țirulnic Serghei Mihai	XI	Colegiul Național „Grigore Moisil”	serghei.tirulnic@gmail.com				
					Danilov Alexandru-Cristian	XII	Colegiul Național „Grigore Moisil”	alexandru.danilov2005@gmail.com				

Elev 3-argument*

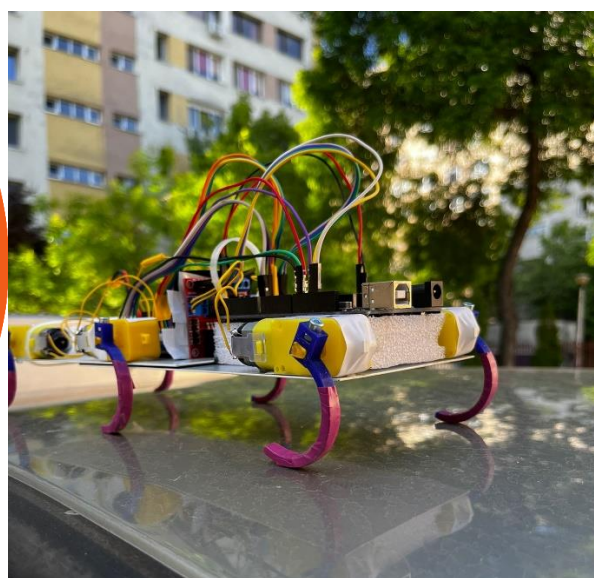
Declarație

Elevul Danilov Alexandru - Cristian împlinește vârsta de 19 ani pe 30 august 2024, dar prezența sa este imperios necesară în echipă deoarece a avut contribuții semnificative și a coordonat versiunea anterioară a proiectului.

Coordonatorul proiectului certifică prin prezenta că declarațiile și informațiile de mai sus sunt corecte, că s-a luat act de prevederile regulamentului concursului, că proiectul a fost finalizat în anul școlar în curs și că se intenționează participarea la Olimpiada de creativitate științifică.

București, 2024

ATREX 2.0 - All Terrain Robotic Explorer



Cuprins

Capitolul I. Utilitate practică	4
Capitolul II. Mecanica	6
1.Complexitate	6
2.Eficiența în construcție	7
Capitolul III. Electronică.....	8
Capitolul IV. Software.....	9
Capitolul V. Design industrial	11

Capitolul I. Utilitate practică

Scopul

Scopul robotului ATREX 2.0 este de a oferi o soluție eficientă și sigură pentru explorarea și supravegherea terenurilor accidentate sau periculoase pentru oameni. Prin utilizarea sa în diferite scenarii de misiuni de recunoaștere și explorare, ATREX 2.0 își propune să reducă riscurile pentru personalul uman și să ofere date precise și utile pentru luarea deciziilor sau în diferite experimente științifice.

Obiective

Principalele obiective ale robotului ATREX 2.0 sunt:

- **Navigare pe teren accidentat:** Capacitatea de a traversa terenuri dificile și de a depăși obstacolele pentru a ajunge în zonele inaccesibile pentru oameni.
- **Recunoaștere și supraveghere:** Utilizarea senzorilor și a camerei pentru identificarea și monitorizarea obiectelor și a persoanelor în diverse medii și condiții.
- **Autonomie și control eficient:** Dezvoltarea unui sistem robust de control și autonomie care să permită robotului să funcționeze eficient și să execute sarcini predefinite fără intervenția umană constantă.
- **Flexibilitate și ușurință de întreținere:** Asigurarea unui design modular și accesibil care să faciliteze modificările, reparațiile și întreținerea robotului în teren.

Problema identificată spre rezolvare

Principala problemă identificată care a condus la dezvoltarea robotului ATREX 2.0 este necesitatea de a explora și monitoriza terenuri periculoase sau greu accesibile pentru oameni fără a expune personalul uman la riscuri inutile. De exemplu, în cazul dezastrelor naturale sau al altor situații de urgență, este crucial să se obțină informații rapide și precise despre situație pentru a putea lua decizii informate și pentru a oferi ajutorul necesar. ATREX 2.0 vine ca o soluție pentru această problemă, oferind o soluție sigură și eficientă pentru explorarea și supravegherea zonelor periculoase.

Descrierea tehnică a robotului ATREX

ATREX 2.0 (All Terrain Robotic Explorer) este un robot hexapod conceput pentru a opera pe teren accidentat. Cu o greutate redusă și un design agil, acesta poate naviga eficient pe terenuri dificile și

poate fi echipat cu o gamă variată de senzori și instrumente, inclusiv o cameră capabilă să recunoască oameni și obiecte.

Structură și componentele principale

Robotul ATREX 2.0 este construit pe baza unui suport hexapodic, având șase picioare articulate, fiecare fiind echipat cu motoare și senzori pentru a permite mișcări precise și stabilizare pe teren accidentat. Carcasa sa este fabricată din materiale ușoare, dar rezistente, care îi conferă agilitate și capacitate de manevrare.

Componentele principale ale robotului includ:

- **Motoare și drivere:** Fiecare picior al robotului este propulsat de motoare cu curent continuu și este controlat de drivere de motor care asigură mișcări precise și adaptabilitate la diferite condiții de teren.
- **Senzori:** ATREX 2.0 poate fi echipat cu o serie de senzori, inclusiv senzori de proximitate, senzori de inerție.
- **Cameră:** O cameră montată pe robot oferă posibilitatea de a recunoaște obiecte și a oameni, permițându-i să îndeplinească sarcini de supraveghere și explorare.

Controlul și autonomia

Robotul ATREX 2.0 poate fi controlat de la distanță prin intermediul unui smartphone sau laptop. De asemenea, poate fi utilizat în mod autonom, putând executa sarcini fără intervenția umană.

Software și comunicații

Controlul robotului se realizează prin intermediul unui software special, care gestionează mișcările, colectarea și procesarea datelor de la senzori și interacțiunea cu mediul înconjurător. Comunicarea cu robotul se poate face prin diverse modalități, inclusiv conexiuni wifi.

Performanțe și capacități

ATREX 2.0 are capacități impresionante de explorare și navigare pe terenuri accidentate și în medii periculoase pentru oameni. Prin designul său robust și capacitatea de a depăși obstacolele, acesta poate înlocui un om trimis într-o misiune de recunoaștere sau explorare în zone greu accesibile.

Exemple de utilizare:

- **Misiuni de recunoaștere:** Robotul poate fi folosit pentru a explora și cartografia zonele afectate de dezastre naturale sau de alte evenimente care pun în pericol viețile oamenilor.
- **Explorare în medii periculoase:** Capacitatea sa de a opera în medii cu temperaturi extreme, condiții meteorologice dificile sau chiar în zone contaminate radioactiv este extrem de valoroasă pentru activitățile de supraveghere și cercetare.

Flexibilitatea și ușurința de întreținere

Designul modular al robotului ATREX 2.0 îi conferă o mare flexibilitate și ușurință în întreținere și modificare. Componentele sale pot fi înlocuite rapid și eficient, iar accesibilitatea electronicii permite efectuarea de teste și depanări într-un timp scurt.

Costurile și disponibilitatea pieselor

Cu un cost relativ redus de aproximativ 250 de lei și utilizarea unor piese comune și accesibile, robotul ATREX 2.0 este o soluție economică și practică pentru diverse aplicații de explorare și supraveghere. Piese necesare pentru construcția și întreținerea sa sunt disponibile pe scară largă, ceea ce facilitează repararea și modernizarea acestuia.

Capitolul II. Mecanica

1.Complexitate

Complexitatea mecanică este dată de cele 6 picioare flexibile, acționate în mod direct de 6 motoare cu reductor. Acestea, împreună cu partea electronică, sunt montate pe o placă metalică.

Robotul folosește 6 motoare de curent continuu (Fig.1) cu reductor încorporat, acționarea fiind una de tip direct drive (picioarul este montat direct pe axul care iese din reductor):



Fig.1 Motor de curent continuu

Montarea piciorului este simplă, datorită piciorului printat 3D (Fig.2), care folosește un șurub M3x8 și o piuliță M3 pentru a se strânge pe ax (Fig.3).

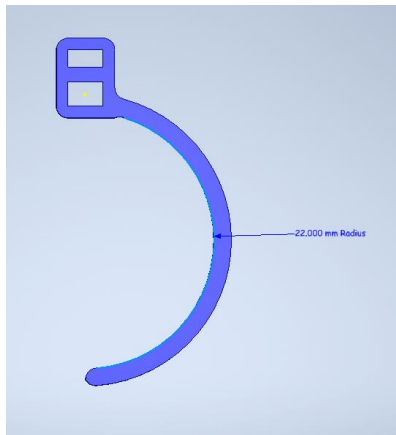


Fig.2 Picior robot

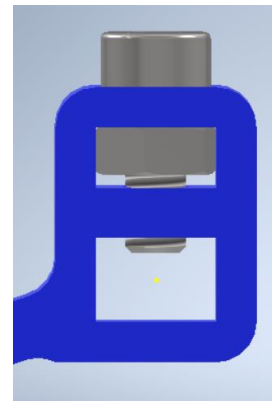


Fig.3 Șurub M3x8 și piuliță M3

2.Eficiența în construcție

Robotul a fost realizat cu ajutorul printării 3D, motiv pentru care am avut nevoie de un software de tip CAD și de un slicer.

Software CAD : AutoDesk Inventor (acesta a fost folosit atât pentru proiectarea pieselor printate, cât și pentru simularea picioarelor)

Picioarele au fost printate din PLA, iar dimensiunile importante (grosimea și lățimea) au fost stabilite folosind FEA, integrat în Inventor.

Folosind o încărcătură aproximativ egală cu masa robotului, pe un singur picior obținem valori mici ale deformării și un factor de siguranță suficient. Pentru simulări folosim Polistiren ca substituent PLA, ABS ca material al axului de la motor.

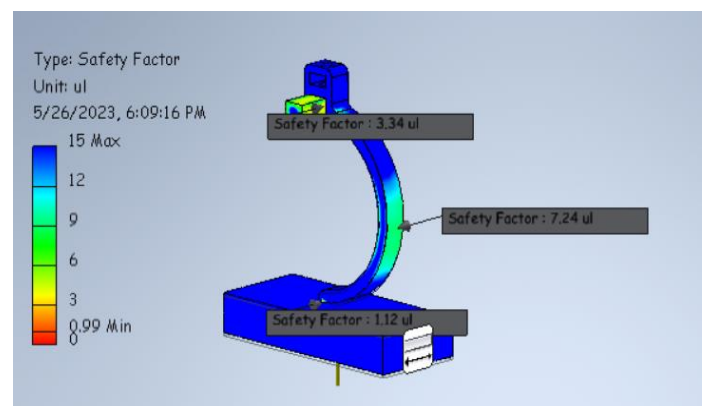
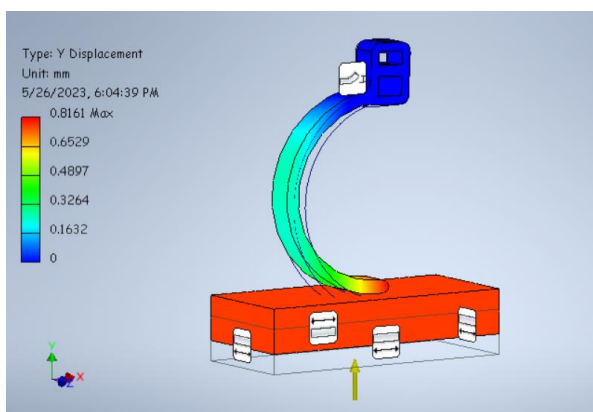


Fig.4 Deplasarea pe direcția verticală a piciorului, la încărcătură de 8N

Fig.5 Factor de siguranță, cu sonde plasate în locurile de interes.

Figura 5 prezintă factorul de siguranță. Când robotul se deplasează pe o suprafață dură, vârful piciorului este supus unui efort crescut, motiv pentru care am ales să folosim „păpucei” din burete. Cazul prezentat este unul extrem, în care toată greutatea robotului este susținută de unul din cele 6 picioare.

Capitolul III. Electronică

Partea electronică are ca scop controlarea motoarelor folosind un MCU.

- 1) Sursa de energie: pentru teste am folosit o sursă de tensiune din laboratorul de fizică, dar am trecut ulterior la o baterie. Bateria a fost aleasă după mai multe criterii: tensiune apropiată de tensiunea nominală a motoarelor, greutate redusă, asigurarea unei autonomii de cel puțin 10 minute. Folosim o baterie LIPO 2S, 400mAh. ($2 \times 3.7V = 7.4V$) .
- 2) Motoarele : folosim 6 motoare de curent continuu, cu tensiune nominală de 6V.
- 3) Drive de motor: Deși arhaice, driverele L298N s – au dovedit a fi perfecte pentru proiect. Fiecare dintre acestea permite controlul individual al direcției și al vitezei a două motoare, folosind semnale PWM. Mai mult decât atât, acestea au un coborâtor de tensiune încorporat, care ne permite să reducem tensiunea la 5V, pentru a alimenta MCU – ul. Folosim 3 astfel de drivere (cu breakout board)
- 4) Arduino MEGA : Controlând 6 motoare prin 3 drivere, folosim 18 pini GPIO, dintre care 6 capabili de PWM, cum nu am dorit să folosim un multiplexor, acest MCU ne oferă toți acești pini. Poate fi alimentat direct la 5V.



Fig.6.Acumulator LIPO



Fig.7Arduino MEGA

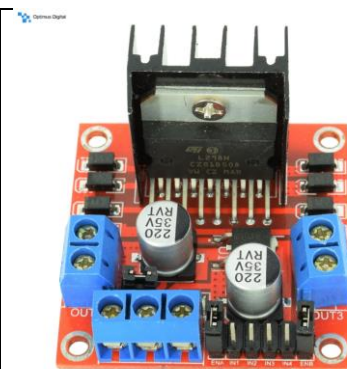


Fig.8. Drive de motor

Capitolul IV. Software

Robotul este controlat de un microcontroller, Arduino Mega și un Raspberry Pi 3. Arduino controlează motoarele, codul este scris în C++ și cuprinde 4 funcții principale: mers înainte, mers în stânga, mers în dreapta și oprire.

Pentru o bună sistematizare a firelor am decis să includem la începutul codului o „hartă” a electronicelor: Această abordare facilitează urmărirea și înțelegerea codului, oferind o referință vizuală clară a conexiunilor fizice.

```
void m0()
{
    analogWrite(13, abs(250));
    {
        digitalWrite(53, HIGH);
        digitalWrite(52, LOW);
    }
} //actioneaza motorul cu viteză mare

void s0()
{
    analogWrite(13, abs(100));
    {
        digitalWrite(53, HIGH);
        digitalWrite(52, LOW);
    }
} //actioneaza motorul cu viteză mică

void stop0()
{
    analogWrite(13, abs(0));
    {
        digitalWrite(53, LOW);
        digitalWrite(52, LOW);
    }
} //oprește motorul
```

Funcțiile de control pentru motorul 0

```
void inainte()
{
    m0();
    m1();
    m2();
    m3();
    m4();
    m5();
} //robotul se deplasează în față

void stanga()
{
    m0();
    s1();
    m2();
    s3();
    m4();
    s5();
} //robotul virează la stânga

void dreapta()
{
    s0();
    m1();
    s2();
    m3();
    s4();
    m5();
} //robotul virează la dreapta
```

Funcțiile pentru deplasare

Pentru a putea controla robotul de la distanță și pentru a putea implementa și o autonomie, am folosit un Raspberry Pi 3 dotat cu o cameră. Codul din Raspberry Pi este scris în Python și, cu ajutorul unei conexiuni prin serial la Arduino, returnează apeluri ale funcțiilor care controlează direcția robotului. Astfel, robotul poate funcționa controlat direct de utilizator, sau autonom.

În autonomie, robotul identifică oameni și determină în ce direcție trebuie să meargă robotul pentru a ajunge la aceștia în funcție de poziția lor în cadru. Acest lucru este posibil folosind cv2 și descriptorul HOG împreună cu câteva marcaje asupra cadrului pentru a facilita identificarea și orientarea.

```
while True:
    for frame in camera.capture_continuous(rawCapture, format="bgr", use_video_port=True):
        image = frame.array
        cv2.line(image, (263, 0), (263, 480), (204, 0, 204), 2)
        cv2.line(image, (376, 0), (376, 480), (204, 0, 204), 2)
        # cele 2 linii de culoare violet delimiteaza imaginea pentru a determina daca omul se afla in stanga/dreapta/mijlocul imaginii
        gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
        boxes, weights = hog.detectMultiScale(image, winStride=(8,8) )
        boxes = np.array([[x, y, x+w, y+h] for (x, y, w, h) in boxes])
        for (xA, yA, xB, yB) in boxes:
            cv2.rectangle(image, (xA, yA), (xB, yB), (204, 0, 204), 2)
            # chearul de culoare violet apare in jurul omului
            cv2.line(image, (round((xA+xB)/2), 0), (round((xA+xB)/2), 480), (0, 153, 51), 2)
            # linia de culoare verde traverseaza axa de simetrie verticala a chenarului pentru a determina mai usor pozitia omului
            if round((xA+xB)/2) <= 263:
                print("stanga")
                ser.write(b"stanga\n")
            if round((xA+xB)/2) > 263 and round((xA+xB)/2) < 376:
                print("inainte")
                ser.write(b"inainte\n")
            if round((xA+xB)/2) >= 376:
                print("dreapta")
                ser.write(b"dreapta\n")
            # se aviseaza in terminal si se scrie in serial pozitia curenta a omului
        cv2.imshow("Frame", image);
        key = cv2.waitKey(1) & 0xFF
        rawCapture.truncate(0)
        if key == ord("o"):
            print("OVERRIDE")
            break
    # la apasarea tastei 'o' se iese din autonomie, astfel robotul poate fi controlat prin tastatura
```

Recunoastere umana

```
void loop()
{
    if(Serial.available()) ///se acceseaza conexiunea prin serial cu Raspberry Pi 3 de unde robotul primeste directia de deplasare
    {
        command=Serial.readStringUntil('\n');
        command.trim();
        if(command.equals("stanga"))
            stanga();
        if(command.equals("inainte"))
            inainte();
        if(command.equals("dreapta"))
            dreapta();
        if(command.equals("opreste"))
            opreste();
    }
}
```

Citire comenzi primite prin serial 1

```
if __name__ == '__main__':
    ser=serial.Serial('/dev/ttyUSB0', 9600, timeout=1) #se verifica canalul cu ls /dev/tty*
    ser.flush()

# conexiune prin serial între Arduino mega si Raspberry Pi ('/dev/ttyUSB' poate fi urmat de 0 sau de 1 în functie de portul USB în care se afla conexiunea.)
# Pentru a afla portul corect se ruleaza în terminal comanda $ ls/dev/tty*
```

Initializare comunicare serial Raspberry 1

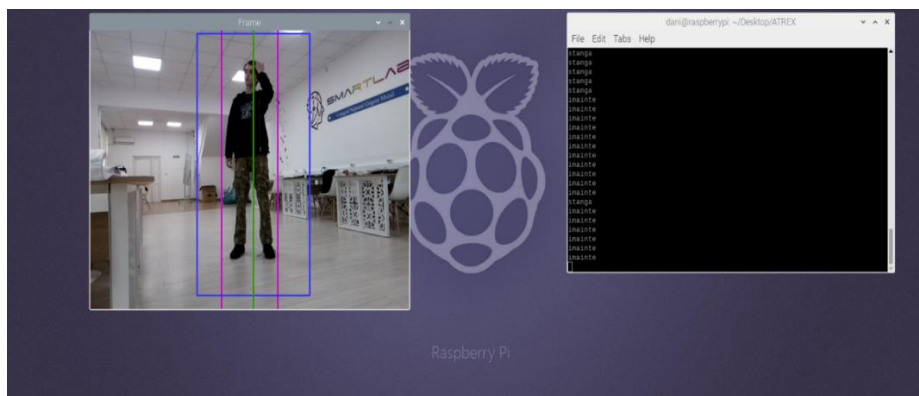


Fig.9.Recunoaștere de persoană folosind CV2

În plus, autonomia are și o funcție de override, care permite utilizatorului să preia controlul robotului de la tastatură în cazul apariției unei erori neprevăzute sau a unei situații critice. Această funcție asigură că utilizatorul poate interveni și să preia controlul direct asupra robotului pentru a evita orice pericol sau pentru a corecta eventuale probleme. Astfel, în caz de nevoie, utilizatorul poate să opereze manual robotul folosind tastatura, chiar dacă acesta se află în modul de autonomie.

```

break
# la apăsarea tastei 'o' se iese din autonomie, astfel robotul poate fi controlat prin tastatura
while True:
    char = screen.getch()
    # se capteaza inputul de la utilizator
    if char == curses.KEY_UP:
        print("inainte")
        ser.write(b"inainte\n")
    if char == curses.KEY_LEFT:
        print("stanga")
        ser.write(b"stanga\n")
    if char == curses.KEY_RIGHT:
        print("dreapta")
        ser.write(b"dreapta\n")
    if char == curses.KEY_DOWN:
        print("opreste")
        ser.write(b"opreste\n")
    # in functie de input se afiseaza in terminal si se transmite prin serial directia in care va merge robotul
    if key == ord("c"):
        break
if key == ord("c"):
    break

```

Capitolul V. Design industrial

Cu cele 6 picioare ale lor, roboții de tip RHEX au doar 6 motoare, care, în ciuda simplității mecanice îi asigură o stabilitate intrinsecă și posibilitatea de locomoție complexă.

Piese de plastic sunt concepute astfel încât să fie ușor printabile 3D și ușor de înlocuit. Toate cele 6 picioare se printează în sub o oră. În procesul de proiectare am încercat mai multe geometrii ale piciorului, dar am rămas la o formă semicirculară, ale cărei dimensiuni le – am determinat prin simulări.

Pentru configurarea design-ului acestui robot au fost folosite cunoștințe de mecanică, geometrie, desen tehnic, electronică, informatică, proiectare. Astfel, putem considera că proiectarea robotului asigură utilizarea și reproducerea acestuia cu succes și la scară mare.

BIBLIOGRAFIE

1. Abhishek Anand Bapat, *Design, Prototyping And Testing of an Autonomous Hexapod Robot With C Shaped Compliant Legs: Abhishek*, The University of Texas at San Antonio Department of Mechanical Engineering, december 2016
2. Haynes G. Clark, Jason Pusey, Ryan Knopf, Aaron M. Johnson, and Daniel E. Koditschek, *Laboratory on Legs: An Architecture for Adjustable Morphology with Legged Robots*, University of Pennsylvania, 2012
3. Uluc Saranli, Martin Buehler, Daniel E. Koditschek, *RHex: A Simple and Highly Mobile Hexapod Robot*, The International Journal of Robotics Research Vol. 20, No. 7, pp. 616-631, Sage Publications, 2001
4. Pei-Chun Lin, Haldun Komsuoglu, Daniel E. Koditschek, *Sensor Data Fusion for Body State Estimation in a Hexapod Robot With Dynamical Gaits*, IEEE Transactions on Robotics, University of Pennsylvania, Volume 22, Issue 5, 2006
5. Kevin C. Galloway, G. C. Haynes, B. Deniz Ilhan, Aaron M. Johnson, Ryan Knopf, Goran Lynch, Benjamin Plotnick, Mackenzie White, D. E. Koditschek, *X-Rhex, A Highly Mobile Hexapedal Robot for Sensorimotor Tasks*, november 4, 2010

WEBOGRAFIE

1. <https://kodlab.seas.upenn.edu/past-work/rhex/edubot/> (accesat la data de 10 februarie 2024)
2. <https://kodlab.seas.upenn.edu/past-work/rhex/sandbot/> (accesat la data de 12 februarie 2024)
3. <https://www.youtube.com/watch?v=806rzXWYhPg> (accesat la data de 10 martie 2024)
4. <https://www.youtube.com/watch?v=ISznqY3kESI> (accesat la data de 12 martie 2024)
5. <https://www.youtube.com/watch?v=VL0aiQAm4RU> (accesat la data de 12 martie 2024)