

Fa-te mucificatorul - Georgina Potîng

Introducere

Proiectul presupune un Drinking Game pentru a ridica spiritele studenților după sesiune. Combină mai multe jocuri ca și:

- Trivia
- Truth or Dare
- Never have I ever

Într-un singur mecanism.

Scopul este ca jucătorii să iasă din zona de confort, să se descopere unul pe celălalt, să lege prietenii de sânge și, **evident**, să se facă muci.

Am ales să fac acest proiect pentru că să combin activitățile mele preferate: board games și petrecutul cu băutură. Știu sigur că proiectul ăsta va fi utilizat și înafara materiei de PM.

Perfect pentru cercurile de prieteni ai prietenilor, când nu-i cunoști pe restul destul de bine și vrei să ieși din awkward silence.

Descriere generală

Cum am zis, proiectul e un Drinking game.

Flow-ul este următorul:

1. Se apasă un buton care activează servomotorul și se oprește random la un jucător.
2. Se apasă al doilea buton care afișează pe LCD una din mai multe categorii de jocuri Random: truth or dare, physical challenge, etc.
3. Se apasă un al treilea buton care afișează pe LCD întrebarea/provocarea în sine.

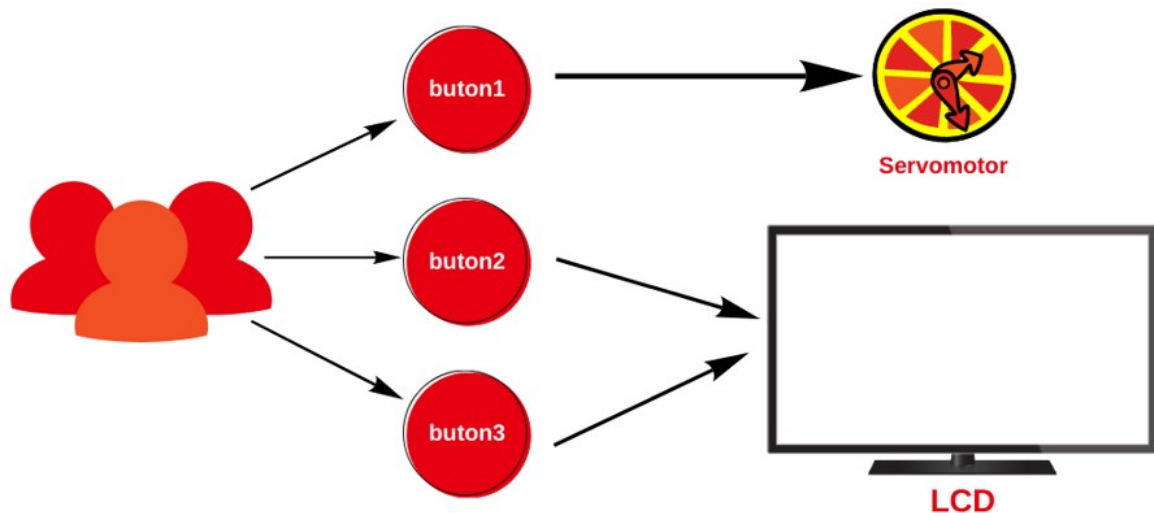
Practic, dacă jucătorul alege să facă provocarea/ să răspundă la întrebare, se trece la următorul jucător.

Dacă nu face provocarea/răspunde la întrebare, va trebui să bea, după care se trece la următorul jucător.

De asemenea, o să am un buzzer care scoate sunete la intervale relativ random, semnal pentru

jucatori sa bea. Ultimul care bea va trebuie sa mai bea inca o dată - ca să nu se plictisească restul.

Schema bloc



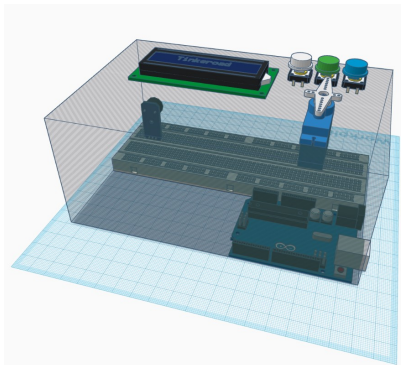
Inputul vine din exterior, de la jucători. Aceștia activează butoanele pentru ca să activeze prin intermediul Arduino servomotorul și LCD-ul. Servomotorul decide următorul jucător și LCD-ul afișează provocările jucătorilor.

Hardware Design

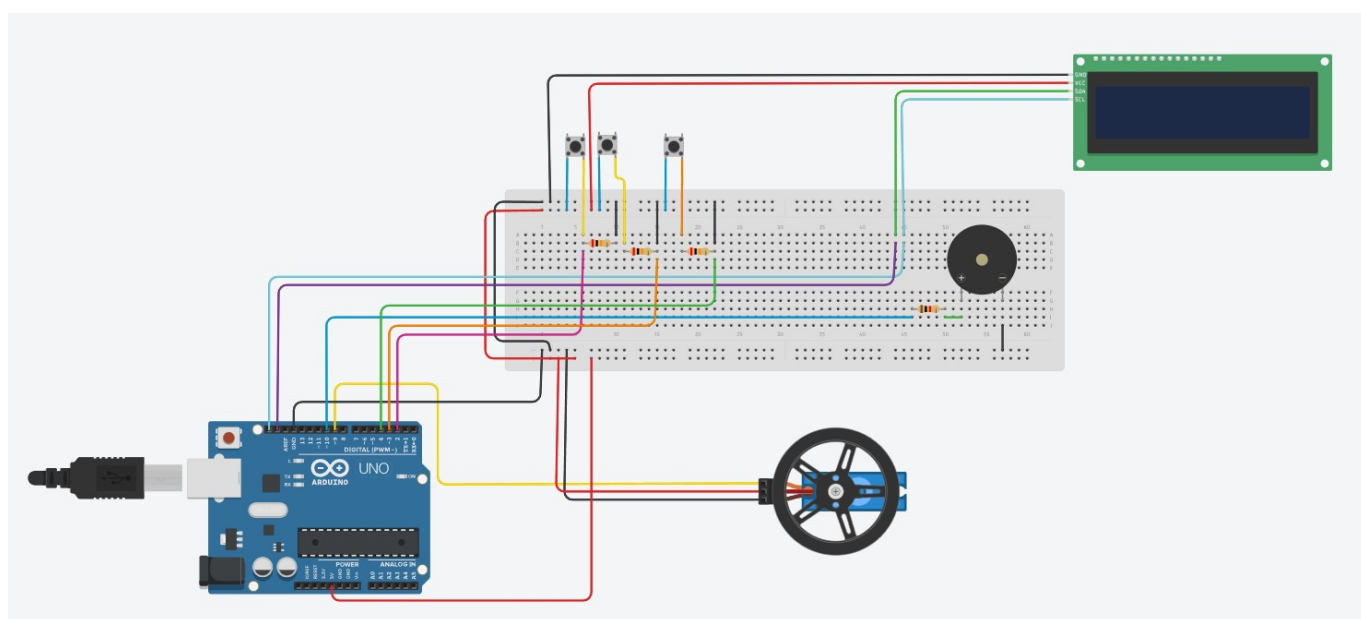
Piese folosite:

- Arduino UNO
- Buzzer
- LCD 20×4
- x3 Butoane
- Servomotor

Design-ul cutiei 3d:



Schema circuitului:



Software Design

Dezvoltarea codului am realizat-o in Arduino IDE (care ar putea fi mult mai imbunatatit)

Pe partea de software am folosit bibliotecile:

1. LiquidCrystal_I2c.h
2. Servo.h
3. Wire.h

Pentru generarea categoriilor am creat un array bidimensional numit `categories[][]`, in care primul element e marimea sirului de caractere.

Pentru fiecare categorie am creat alte tabele, urmand exemplul celui de mai sus.

Am setat ca pini de input butoanele, iar ca output Servomotorul si LCD-ul.

Deoarece am avut nevoie de mai multe generari de numere random, am cautat o metoda de a crea

aceste numere cu un seed de asemenea random: conectand la pinul analog A0 un fir lasat in aer → iar valoarea aceasta sa fie folosita pentru `randomSeed()`.

Funcția `randomFunction()` ia ca parametru un int de la 1 la 3 (pentru cele 3 cazuri de butoane) și în cadrul switchului returnează valori pentru toate butoanele. De exemplu: dacă vreau să se învarte servomotorul pentru un timp random (de la 1 la 3 secunde), funcția returnează un număr de la 1 la 5. Dacă apăs butonul de generare de categorii → se returnează un număr de la 1 la 5.

Aceste rezultate sunt ulterior tratate în loop pentru delay-ul servomotorului și indexul întrebărilor.

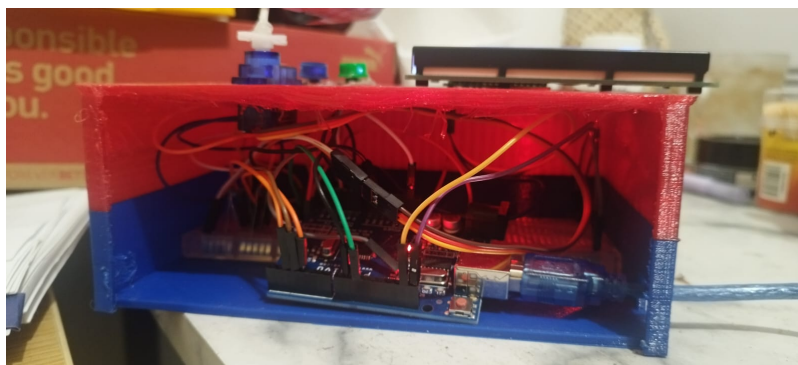
Pentru afișarea întrebărilor a trebuit să mai fac o funcție, `[b]chooseQuestion(int category, int questionIndex)[/b]`, care leagă practic indexul categoriei cu categoria de întrebări în sine și printează la LCD întrebarea.

În afara de funcționarea principală, am adăugat niște tonuri (la fel, random) în loop. Sunetul porneste la un interval între 1 și 3 minute: semnal pentru jucători să dea un shot :)

Funcțiile folosite în cod:

Rezultate obtinute

Cutia interioara =)



Cutia exterioara 8-o



Jurnalul proiectului:

- 28.04.2023 - Am finalizat ideea de baza a jocului, am creat flow-ul si am ales piesele potrivite pentru circuit
- 11.05.2023 - Am comandat piesele pentru proiect.
- 12.05.2023 - Mi-au venit piesele pentru hardware. Am realizat ca am uitat sa cumpar breadboard, adaptor I2C si servomotor, asa ca am mai plasat o comanda la emag.
- 19.05.2023 - Am lipit pinii de la LCD la adaptorul I2C si firele tata-tata la butoane.
- 24-26.05.2023 - !! M-am chinuit sa inteleg de ce nu-mi printa nimic LCD-ul, desi backlight-ul pornea, am refacut circuitul, refacut codul, re-descarc arduino IDE, sters si redescarcat biblioteca LCDI2C → CA SA MI SE SUGEREZE CA APARENT TREBUIA DOAR SA REGLEZ CU SURUBELNITA POTENTIOMETRUL DIN SPATELE LCD-ULUI !!
- 27.05.2023 - Am ajuns la o versiune decenta a codului.

Concluzii

Deși proiectul nu pare să fie foarte complex nici din punct de vedere software și nici hardware, am pus accentul pe design și pe funcționalitate. Am vrut să creez un joc care să fie eventual folosit, îmbunătățit și care să rămână ca amintire :)

Probleme am intampinat si la crearea schemei 3d si printarea acesteia (am masurat prea la fix, nu incape cum trebuie inaptrunici placa nici breadboardul, nici lcd-ul; iar de printat ar fi trebuit sa printez gaurile acesteia pe laturi, nu pe fata de sus a paralelipipedului, fiindca imprimanta nu mi-a facut o baza si practic printa in aer, iar acum inaptrun e foarte urat).

Am petrecut mult timp pe partea de debugging în cod și am căutat soluții pentru probleme de genul: Arduino nu lucreaza bine pe stringuri, are memorie foarte limitată RAM.

Download

Codul cutiei magice: [codginapmzip.zip](#) PDF: [Export to PDF](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/fstancu/georgina>



Last update: **2023/05/29 21:40**