

ECOnomic H2O

(posem fi a les ampolles d'aigua de plàstic)



INS Pere Ribot

Alumna: Carla Claramunt

Tutora TR: Maria Suárez

2n Bat B

20/12/2018

AGRAÏMENTS

Gràcies a la Maria Suárez, per haver estat el meu suport i empenta no només en aquest projecte. Per ajudar-me a descobrir la meva millor versió per reflectir en el treball. Per la seva dedicació i més.

Gràcies al Jordi Pocurull per despertar-me la curiositat al món de la programació. Però sobretot agraeixo la teva confiança i suport incondicional, gràcies per sempre obrir-me portes al futur.

Gràcies papa per ensenyar-me que la constància és la clau de l'èxit i que no existeix problema sense solució. Gràcies a la teva empresa Frapont per facilitar-me el tall de peces amb maquinària especialitzada fora del meu abast. I gràcies mama per ser un pilar, pel suport moral i la motivació que sempre em saps transmetre.

Per últim també vull agrair a Enrique Prieto i a l'Ajuntament la seva dedicació al projecte, gràcies pel vostre granet de sorra.

Índex

AGRAÏMENTS	2
INTRODUCCIÓ.....	4
PRESENTACIÓ	6
CONTINGUTS DE ROBÒTICA	13
GLOSSARI	56
LA MÀQUINA EXPENEDORA	22
REALITZACIÓ DE LA MAQUETA.....	23
CONSTRUCCIÓ DE L'ESTRUCTURA.....	23
MATERIALS I EINES	23
PROCÉS DE CONSTRUCCIÓ	23
PROGRAMACIÓ DE LA MAQUETA	29
MATERIAL NECESSARI.....	29
PROCÉS DE PROGRAMACIÓ.....	30
FINAL PROCÉS DE LA MAQUETA.....	46
FUNCIONAMENT DE LA MAQUETA	47
RESULTATS.....	51
CONCLUSIONS	54

INTRODUCCIÓ

El meu treball de recerca és la unió de dos grans temes; la robòtica i la millora de la distribució de l'aigua per ajudar al medi ambient. Jo volia que el meu treball donés solució a un problema real, que es pogués traslladar a la realitat. Això l'adaptaria a un àmbit molt útil per al meu futur i la carrera que vull cursar. D'aquesta manera la meva motivació també augmentaria considerablement; sempre m'ha entusiasmat trobar solucions innovadores i relacionades amb el món de la robòtica; de fet aquesta seria la meva feina ideal. És per això que també m'aportaria un alt grau d'experiència i aprenentatge molt útil en el meu futur. Així doncs vaig pensar que la millor manera de fer-ho era mitjançant dos temes que tant m'interessaven com són la robòtica i el medi ambient. Aquest interès es va iniciar fa uns anys quan a robòtica vam fer un projecte sobre el reciclatge. Gràcies al qual vam empatitzar molt més amb la problemàtica. En el meu cas un cop prenc consciència d'un problema m'apareix la necessitat de trobar una solució.

D'altra banda la passió per la robòtica va començar cinc anys enrere. Des de llavors, faig robòtica i participo en la First Lego League amb un grup de Vilassar de Mar. La First Lego League és una competició de robòtica molt completa, ja que combina un joc de robot, en el qual tu has d'haver programat un robot per tal que faci una sèrie de proves en dos minuts i mig; una presentació de valor i un projecte científic. Gràcies a aquesta competició hem tingut la sort de viure l'experiència de finals estatals i una internacional. Allí t'adones compte de la importància del tema de la temporada i de la investigació sobre aquest. També pots veure com varien les solucions segons la cultura o el país. Això també t'enriqueix culturalment i professionalment, ja que coneixes als que poden ser futurs enginyers.

A partir d'aquí la meva passió per la robòtica i aquestes ganes de trobar solucions a problemes que se't presenten van créixer enormement. Aquesta experiència m'ha ajudat a apropar-me una mica més al meu futur professional, a fer camí, a veure què m'agrada i amb què em veig treballant en un futur, a conèixer les meves habilitats i a aplicar tot això en el meu treball de recerca. Per això aquesta

competició ha estat un pilar fonamental pel meu Treball de Recerca, d'allà va néixer la idea.

Jo vaig triar l'innovació a través de la tecnologia en el medi ambient, ja que crec que és un tema en el qual ens queda molt recorregut i del qual em venia de gust informar-me. Vaig decidir el tema de l'aigua, ja que l'any passat el tema de la First Lego League va ser aquest. Amb el tema esmentat vam arribar a la final internacional a Estònia, i no volia que es quedés allà, ho volia portar més enllà. Volia desenvolupar aquest projecte com si hagués de ser dut a terme. Em vaig enamorar del projecte.

Amb aquests dos interessos per sobre em vaig proposar de veure si era capaç de construir una màquina expendedora d'aigua, amb ampolles de vidre per tal d'eliminar les ampolles de plàstic.

Per fer-ho la meva intenció és crear una màquina en la qual tinguessis l'opció de comprar l'ampolla de vidre i de retornar-la. L'aigua seria gratuïta i refrigerada.

PRESENTACIÓ

Tot i la importància de l'aigua per la salut, quan sortim al carrer l'accessibilitat és molt limitada; les fonts no tenen un punt de vista higiènic a causa de la falta de neteja. Estan situades, normalment, prop d'un parc, on els nens i els gossos les embruten amb més freqüència. Tot i la seva posició i el seu ús la neteja d'aquesta no és l'adequada el que pot derivar en infeccions. Tot i ser el principal aquest no és l'únic motiu que tira enrere a l'hora de fer ús de les fonts; també ens trobem que no sempre en disposem una a l'abast, sobretot en llocs tancats. Per últim l'aigua no està refrigerada i a l'estiu això es converteix en un problema més.

L'aigua embotellada és molt cara i molta gent opta per a comprar una beguda ensucrada. La gent no vol pagar per alguna cosa que quan arribi a casa ja tindrà a la seva disposició. Per què pagar per una cosa que gairebé pots obtenir gratuïtament? Però no és saludable per al nostre organisme passar una jornada sense veure aigua. A més el preu psicològic¹ de les begudes ensucrades, encara arrossega més a consumir aquests productes i deixar de beure aigua durant el dia. El preu psicològic utilitza la resposta emocional del consumidor per tal d'augmentar les vendes; fa que la gent compri una beguda ensucrada per la proximitat del seu preu al d'una ampolla d'aigua. A això s'afegeix que a la gent li fa mal gastar-se diners en aigua i no tant en un refresc.

El consum d'aquestes begudes és molt perjudicial i una de les principals causes de l'obesitat; no aporten res més que grasses i sucre. Segons dades de l'Organització Mundial de la Salut (OMS), des de 1975, l'obesitat s'ha triplicat en tot el món, i aquestes begudes en són en part les causants.

Un estudi ens explica que el nostre cervell no està preparat per detectar calories líquides i menys si són ensucrades. No només ens aporten una gran quantitat de calories d'alta disponibilitat sinó que també redueix la sacietat, fent que les persones consumeixin més calories durant el dia. Les multinacionals d'aquest

¹ Preu psicològic: tècnica de màrqueting que utilitza el joc psicològic per a aconseguir més vendes. Per exemple les empreses de les begudes ensucrades aproximen molt el preu al d'una ampolla d'aigua per augmentar les vendes.

sector acostumen a dir que els seus productes contenen 1 kcal. Ens afirmen que una kcal és una kcal, tot i que la ciència en diu que no és el mateix 50 kcal de bròcoli que de qualsevol d'aquestes begudes. Per demostrar-ho es va fer un estudi en el qual persones amb sobrepès van prendre el 25% de les calories en forma de beguda ensucrades durant 10 setmanes, i amb una condició, no passar el nombre de calories recomanades. I el resultat va ser l'esperat, les persones alimentades a base de begudes ensucrades van engreixar i les proves mèdiques van empitjorar.

Les begudes ensucrades s'han associat amb l'increment de l'obesitat i de sobrepès a escala mundial. Són molts els estudis epidemiològics que han trobat una relació entre el consum d'aquestes begudes i aquesta malaltia.

La diabetis mellitus és una malaltia crònica que apareix quan el pàncrees no produeix prou insulina o l'organisme no la utilitza de manera eficaç. Aquesta malaltia cada vegada afecta més malalts a la nostra societat, dades com les de l'OMS o de l'associació internacional de la diabetis ens ho confirmen. Les dades de l'OMS ens ensenyaven que el 1980 el nombre de persones amb diabetis eren 108 milions, en canvi el 2014 aquesta xifra va augmentar fins a 422 milions.

Una altra malaltia que podem associar a aquest producte és la diabetis mellitus tipus 2. Un dels millors estudis realitzats fins ara, l'estudi EPIC, va revelar aquesta relació.

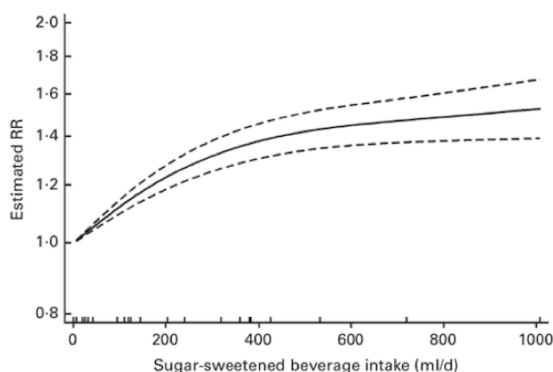


Fig. 2. Non-linear dose-response curve for sugar-sweetened soft drink intake and type 2 diabetes risk. —, Best-fitting cubic spline; ---, 95% CI. RR, relative risk.

La gràfica mostrada ens indica el risc relatiu associat al consum d'aquestes begudes i la diabetis mellitus tipus 2.

Font:Google

Les malalties cardiovasculars també són esclaves d'una societat sedentària i amb uns mals hàbits en l'alimentació. Això es veu accentuat amb les begudes ensucrades, aquest és un dels motius pels quals aquestes malalties van tenint cada vegada més ressò en el nostre dia a dia. Aquests fets s'han anat demostrant amb investigacions, com la de Chun (vegeu annex 3) i alguns col·laboradors; en la que van avaluar a 2200 adults i van arribar a la conclusió que alts nivells de sucre, com els que ens aporten les begudes com la cola, estan estretament relacionades amb tantes malalties com les cardiovasculars, el càncer o la diabetis mellitus tipus 2. A més les dades de l'OMS ens diuen que l'any 2012 per causa de les malalties cardiovasculars van morir 17,5 milions de persones al món.

A sobre l'aparició d'aquestes malalties no està tan relacionat amb el pes sinó amb el consum de productes tan perjudicials com les begudes ensucrades.

A Espanya es consumeixen més de 20 L només de refresc de cola a l'any per capita. Aquest valor es podria reduir si milloréssim l'accessibilitat de l'aigua, ajudant a la gent a tenir un estil de vida més saludable. D'aquesta manera solucionaríem el primer problema detectat. Com aquest problema afecta a tota la societat podria ser finançat pels ajuntaments. Aquests tenen pressupostos per campanyes de consciència sobre la salut, o per promoure hàbits saludables i l'aigua és imprescindible per a la nostra salut.

Nosaltres som aigua, el 70% del nostre cos és aigua. Per això és imprescindible per a mantenir el nostre cos sa, neteja el nostre cos i elimina toxines. És el vehicle ideal, transporta els nutrients (vitamines, minerals i glucosa) i l'oxigen a totes les cèl·lules del nostre cos. El plasma sanguini està format per un 91% d'aigua.

A més l'aigua està present en tots els teixits i en els òrgans vitals del nostre cos: els ronyons, el cervell, el fetge, els pulmons i el cor. És un element fonamental en processos com ara la digestió i l'eliminació de residus metabòlics que no podem digerir. També és qui estructura el sistema circulatori. Una altra funció és la de regular la temperatura corporal, el nostre organisme utilitza l'aigua per a alliberar calor quan la temperatura ambient és elevada; ho fa a través de la suor. L'aigua s'encarrega de lubricar les articulacions; ja que gràcies a l'aigua del líquid sinovial o sinòvia, les nostres articulacions poden funcionar de manera fluida. Per això beure poca aigua pot provocar lesions i artritis.

A més té nombrosos beneficis per a la salut que han estat comprovats amb diversos estudis. L'aigua alleuja la fatiga, això és degut al fet que les toxines i els residus no han estat eliminats; és una funció de l'aigua, i el cor ha de fer un esforç major per tal de bombar la sang a totes les cèl·lules i altres òrgans principals. Però aquest no és l'únic benefici, la falta d'aigua molts cops és la que ens provoca migranyes i mals de cap. A més ajuda a mantenir hidratada la pell, augmenta la seva elasticitat i a regenerar els teixits de la pell. També ens ajuda a fer la digestió i evita el restrenyiment. Només pel simple fet de beure aigua estem reduint el risc de patir alguns tipus de càncer com ara el de còlon i de bufeta o el risc de patir problemes cardíacs. A més ens ajuda a accelerar la pèrdua de pes. Així doncs la substitució de les begudes ensucrades per l'aigua millorarien clarament la salut de la població, reduint molts problemes com ara l'obesitat o les malalties cardiovasculars.

Cal afegir que la distribució de l'aigua és molt poc eficaç quan es fa a través de les ampolles de plàstic, que és gairebé sempre. El plàstic no és un bon aliat per al medi ambient i per tant les ampolles tenen molts inconvenients en molts àmbits:

- La salut: La majoria de les ampolles d'aigua embotellada estan fabricades amb plàstics PET (tereftalat de polietilè), que poden ser perjudicials per a la nostra salut. Aquest plàstic deixa partícules que desprenen a l'aigua i que acabem ingerint. Aquestes partícules són l'antimoni i aquesta ha estat catalogada per la IARC (Agència Internacional per a la Investigació del Càncer) dins del grup 2B; volent dir que causa càncer. Trobem altres substàncies com per exemple el polipropilè o el poliestirè. Aquest fenomen encara és més present quan l'ampolla ha estat exposada a altes temperatures i/o quan perdura molt de temps en el recipient. També cal contradir que no desprèn BPA (bisfenol A) com alguns mitjans de comunicació asseguren.
- El medi ambient: el consum d'energia per la fabricació procedeix del petroli o de materials derivats d'aquest. Cal afegir que la base dels materials també és d'aquesta procedència.

En aquest cas també parlem de l'empremta de carboni, que és proporcional a la quantitat d'energia que es consumeix al produir una unitat del producte. En el cas de les ampolles de plàstic, l'empremta és considerable; principalment degut a la intensitat energètica del seu procés d'elaboració. Així com la baixa taxa de reciclatge que té aquest material. En la UE només es reciclen el 21,3% dels plàstics i el 30% es crema per recuperar energia. Això fa que si es venen 50 bilions d'ampolles de plàstic de litre a Europa en un any, s'acumulin 25 bilions en abocadors o encara pitjor acaben tirats a la natura. Diversos estudis mostren valors que varien entre 173 i 250g de CO2 per litre d'aigua embotellada, el que es tradueix a una empremta 6000 vegades major que la que deixa un litre d'aigua freda.

- El cost: En comprar l'aigua en ampolles es paga bàsicament per l'envàs. És per això que s'acusa les empreses que les comercialitzen de privatització d'un bé suposadament comú, l'aigua.
- El transport: En la majoria de casos aquest producte ve de l'altre extrem de la península el que provoca una emissió molt elevada de CO2.

També cal destacar que per la producció de l'envàs de plàstic es necessita més aigua de la que aquest conté. A aquests nombrosos inconvenients encara cal afegir que l'aigua de l'aixeta és més controlada que l'aigua embotellada.

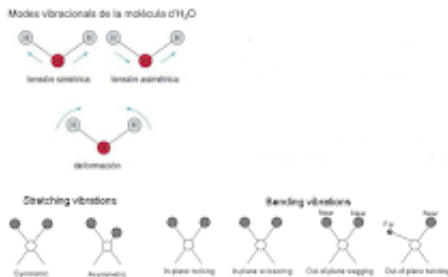


Com aquesta problemàtica està començant a tenir ressò ja han aparegut intents de la seva prohibició per part de la Unió Europea. Veure annex 1.

impacte
economic i
mediambiental de les ampolles de
plàstic

A arran d'aquest problema han sorgit empreses que es preocupen per al nostre medi ambient i que han presentat solucions innovadores.

Flaska és una empresa que va néixer amb la intenció de substituir el plàstic per l'obligació d'intentar trobar una solució més eficient tant per al medi ambient com per a la salut. Flaska produeix ampolles reutilitzables de vidre que ajuden a cuidar de la teva salut, així com el medi ambient. Aquest és un vidre molt gruixut i a més protegit amb un neoprè, silicona, suro o cotó. El vidre es programa utilitzant un procediment especial denominat TPS (Tecnologia de Programació del Silici); aquest aconsegueix transformar l'estructura vibracional de l'aigua continguda. La molècula de l'aigua té una estructura angular, i pot vibrar de diverses maneres, gràcies a la seva estructura. Aquestes vibracions van associades a estats energètics de la molècula d'aigua. Gairebé totes les



Representació vibracions de les molècules de l'aigua. Font: Google

molècules amb enllaços polars poden absorbir fotons de la radiació IR. L'absorció de fotons IR provoca transicions entre nivells d'energia vibracionals dels enllaços de les molècules.

Fa falta transformar aquesta estructura, ja que l'aigua que arriba a

casa a passat per molts àmbits que no són naturals, perquè l'aigua normalment es filtra per les roques i el sòl. Però com aquesta és transportada per conductes, el que provoca és que aquesta no sigui igual que l'aigua del manantial i que les substàncies no naturals amb les quals ha entrat en contacte provoquin un efecte vibracional en l'aigua que també pot deixar empremta durant un cert temps. L'objectiu és aconseguir que l'estructura vibracional de l'aigua s'assembli més a l'estructura de l'aigua de manantial.



Ampolla de vidre de la marca Flaska. Font: Google

Fill & drink és també un projecte nascut a Barcelona amb l'objectiu de promoure les ampolles termo reutilitzables, un altre material molt interessant per a erradicar el plàstic en aquest ús. Cal afegir que amb aquesta solució l'aigua es manté freda 24 hores.

Un estudiant de disseny industrial irlandès a arran d'aquesta problemàtica ha utilitzat les propietats de les algues per a crear ampolles biodegradables fetes a través de pols de les algues.



Ampolla feta amb algues. Font: Google



Aquesta tendència també ha arribat al sector de la restauració, l'empresa Aguakm0 ven ampolles de vidre a establiments que fan de la restauració sostenible tot d'una filosofia, amb una oferta gastronòmica de proximitat, per fer el mateix amb l'aigua. L'aigua que posen els mateixos establiments ha estat prèviament microfiltrada. A més és una empresa implicada amb projectes socials per fer arribar l'aigua potable a totes aquelles localitats que no la tenen disponible.

Ampolla de vidre
de l'empresa
Aguakm0. Font:

Google

rtància de l'aigua i a totes les millores que encara hem de fer cada any ni na una conferència amb aquest tema. És dur a terme a Sofia, Bulgària. On les ments més brillants del planeta presenten els resultats de les últimes investigacions sobre diferents temes sobre l'aigua.

A arran de tota aquesta recerca vaig decidir que volia crear un canvi de mentalitat en la població i fer una màquina expenedora però d'aigua de la xarxa. I per tant he volgut fer una maqueta per intentar mostrar com m'agradaria que fos.

CONTINGUTS DE ROBÒTICA

Existeixen diferents tipus de components electrònics per crear els teus projectes de programació. Primer s'ha de triar amb quina placa voldrem treballar. Aquestes plaques són processadors o plataformes microcontroladores que actuen com a un ordinador i que controlen entrades i sortides. Per projectes maker; la base dels quals és la creació pròpia, bàsicament tenim dues opcions a triar; l'Arduino i la Raspberry Pi. Tot i que també existeixen altres opcions al mercat de funcionament molt similar com ara Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, o fins i tot els LittleBits per als més petits. Tot i que només estudiarem la possibilitat entre Arduino i la Raspberry Pi, les opcions més adaptades a les necessitats de l'estudiant, sobretot pel seu cost.

L'Arduino és una plataforma de creació d'electrònica de codi obert, la qual està basada en hardware i software lliure, flexible i fàcil d'utilitzar. El hardware lliure són els dispositius on les especificacions i diagrames són públics. I el software lliure són els programes informàtics on tothom té accés al codi. Arduino ofereix la plataforma IDE (Entorn de Desenvolupament Integrat), és un entorn de programació amb el qual qualsevol persona pot crear aplicacions per a les plaques Arduino, per això tenen una gran varietat d'usos. L'Arduino és una placa basada en un microcontrolador ATMEGA. Els microcontroladors són circuits integrats en els quals es poden gravar instruccions. El microcontrolador d'Arduino posseeix el que se'n diu una interfície d'entrada. És una connexió en la qual podem connectar a la placa diferents tipus de perifèrics. La informació que entri d'aquests perifèrics es traslladarà al microcontrolador, que s'encarregarà de processar les dades. Per l'altra banda tenim la interfície de sortida, que s'encarrega de portar la informació que s'ha processat a l'Arduino a altres perifèrics. Els perifèrics poden ser pantalles o altaveus, però també poden ser altres plaques o controladors.

La Raspberry Pi, a diferència de l'Arduino, és un ordinador monoplaca o SBC (Single- Board Computer). És un ordinador simple format per un SoC*, CPU*, memòria RAM*, ports d'entrada i de sortida d'àudio i vídeo, connectivitat de xarxa, ranura SD per a l'emmagatzematge, rellotge, una presa per a l'alimentació, connexions per a perifèrics de baix nivell, entre d'altres. Gairebé el mateix que un ordinador comú, excepte perquè la Raspberry Pi no té botó d'encesa. És un ordinador molt econòmic i de baix consum. Generalment executen sistemes operatius basats en Linux i estan molt relacionats amb l'Open Software. Això no obstant, el desenvolupament en si de la Raspberry Pi no és Open Software, a diferència de l'Arduino. A més de ser un mini ordinador incorpora funcions d'electrònica, així com pins GPIO* (Universal Asynchronous Receiver-Transmitter) i SPI (Serial Peripheral Interface), I²C (Inter-Integrated Circuit). També cal destacar que la potència no és la mateixa de la qual disposem en un ordinador convencional. Aquesta s'ha de connectar a un monitor, un teclat i un ratolí, a més del corrent. Entre els dos models he escollit la Raspberry Pi, ja que té l'avantatge de ser un ordinador a diferència de l'Arduino. A més m'ha cridat més l'atenció, ja que mai l'havia utilitzat. Però això no vol dir que l'Arduino no sigui vàlid per aquest projecte.



Placa Raspberry Pi. Font: Google

El llenguatge que utilitza la Raspberry Pi és el Python. Es pot utilitzar en moltes plataformes com Linux, Windows, OS/2, Mac entre d'altres. Es tracta d'un llenguatge de programació d'alt nivell, a més és un llenguatge interpretat de codi, això vol dir que no necessita ser compilat per poder ser executat. Tal que ens

ofereix avantatges com ara la rapidesa del desenvolupament. La seva filosofia de disseny busca llegibilitat en el codi i la seva sintaxi permet als programadors expressar conceptes en menys línies de codi del qual seria possible en altres llenguatges. Python suporta diversos paradigmes de programació, incloent-hi programació orientada a objectes, imperativa i també funcional o procedimental. Un paradigma és una forma de representar i manipular el coneixement. Representa un enfocament particular o filosofia per a la construcció del programari. No és millor un que un altre, sinó que cada un té avantatges i desavantatges. També hi ha situacions on un paradigma resulta més apropiat que un altre. Presenta un sistema dinàmic i una gestió de la memòria automàtica i té una gran i exhaustiva biblioteca estàndard. Les llibreries ens faciliten la programació, ja que són funcions incorporades en el seu mateix llenguatge, que només cal descarregar amb el comando i que ens ajuden a realitzar moltes tasques habituals sense la necessitat de programar-les des de zero. Aquestes també faciliten molt la lectura i interpretació del programa.

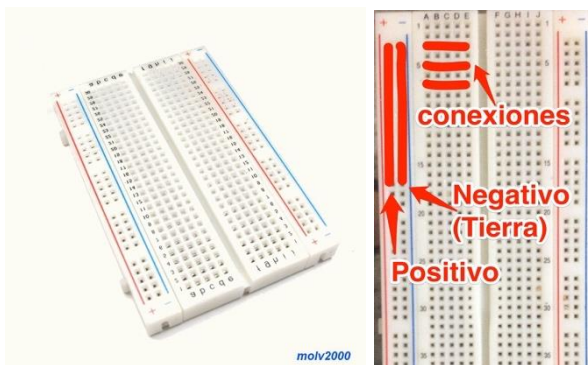
Cal anar amb compte, ja que la Raspberry Pi té descarregat el Python 2 i el Python 3 i algunes llibreries només estan disponibles per un dels dos, o l'escriptura del programa pot tenir alguna petita variació. Per tant és recomanable fer una nota a l'inici del programa indicant amb quin dels dos funciona. Per escriure text o comentaris en un programa només cal afegir un coixinet '#' a l'inici de la frase.

Tornant a l'Arduino, el llenguatge de programació també és diferent, aquest no utilitza el Python sinó el Java. Aquest és un llenguatge orientat a objectes. El Java és un llenguatge interpretat, per això pot semblar lent en comparació amb altres llenguatges, però ofereix un índex de re-utilització de codi molt elevat, sent possible trobar moltes llibreries lliures de Java. És un llenguatge flexible i potent tot i la facilitat amb la qual es programa i dels resultats que ofereix. Un dels trets que el caracteritza i que el fa una eina molt valorada a l'hora de desenvolupar aplicacions distribuïdes, és el fet que és un llenguatge multi-plataforma.

Per connectar els diferents components he utilitzat una Breadboard, que és com un suplement per a facilitar la feina a l'hora de fer les connexions, ja que ens ofereix més espai que la placa. L'ús d'aquest complement pot ser de gran ajuda, ja que la Raspberry Pi com qualsevol placa consta de molt pocs pins i espai per

connectar els elements, el que ens pot fer equivocar a l'hora d'identificar el número de pin.

Es tracta d'una placa que no necessita ser soldada, on es connecten els elements electrònics. Té diferents orificis on es fan les connexions; aquests estan units amb un material conductor. És per això que les connexions s'han de fer en línia recta, tot allò que no estigui a la mateixa fila no estarà connectat. Cal tenir en compte que al mig hi ha un canal que separa la placa per la meitat. Per tant un costat i l'altre no estan connectats almenys que ho fem nosaltres amb un cable conductor.

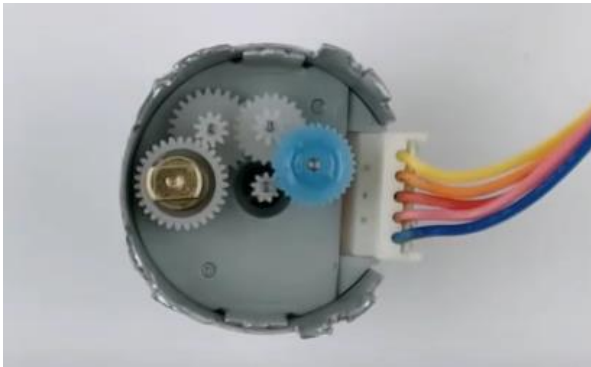


A la Raspberry Pi he connectat diferents elements:

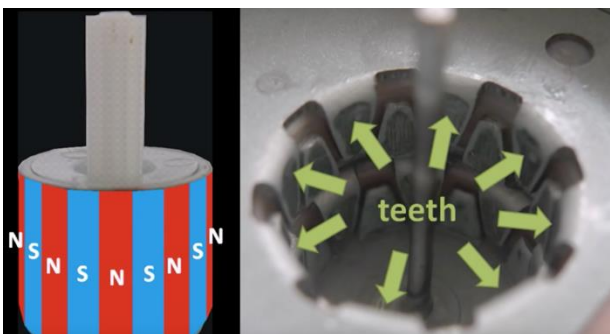
- **Motor:** Els motors són elements de sortida, existeixen diferents opcions de motors, però els principals són els motors de corrent estàndard, els servomotors i els motors pas a pas. Els servomotors respecte als motors de corrent estàndard ofereixen major precisió. L'electricitat és rebuda constantment pels servomotors, i els graus són controlats per un equip de servo regulació. També cal tenir en compte que aquests motors no giren lliurement, sinó que l'angle de rotació es limita a cent vuitanta graus d'anada i de tornada. Però en canvi, el servomotor és capaç d'aguantar esforços i mantenir-se en la posició desitjada un cop a acabat de fer el gir. Un servomotor utilitza una rotació contínua, controlant el gir amb un circuit de control integrat, mentre que el motor pas a pas utilitza diversos electroimants amb dents per executar el moviment. Aquest és molt precís i no necessita un mecanisme de retroalimentació i circuits de suport per a l'accionament de posicionament, un motor pas a pas té el control de posició a través de la seva naturalesa de rotació per increments fraccionals.

Jo he decidit utilitzar el motor pas a pas per les seves qualitats i pel seu funcionament diferents, dins d'aquest tipus utilitzaré el model 28BYJ-48, de 5V. El motor pas a pas és un dispositiu electromecànic que converteix un seguit d'impulsos elèctrics en desplaçaments angulars. Aquests tipus de motors posseeixen l'habilitat de quedar-se clavats en una posició si una o més bobines està amb energia o bé totalment lliure de corrent. A més tenen una gran precisió de gir.

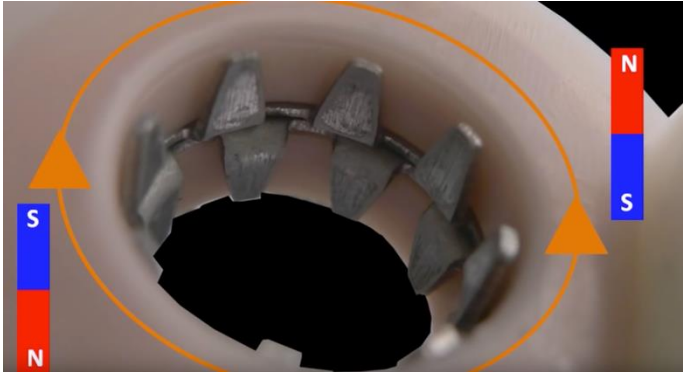
Està format per un sistema d'engranatges per aconseguir una reducció d'1/64, el que vol dir que mentre l'eix de sortida fa una volta el primer que és el central n'haurà de fer 64.



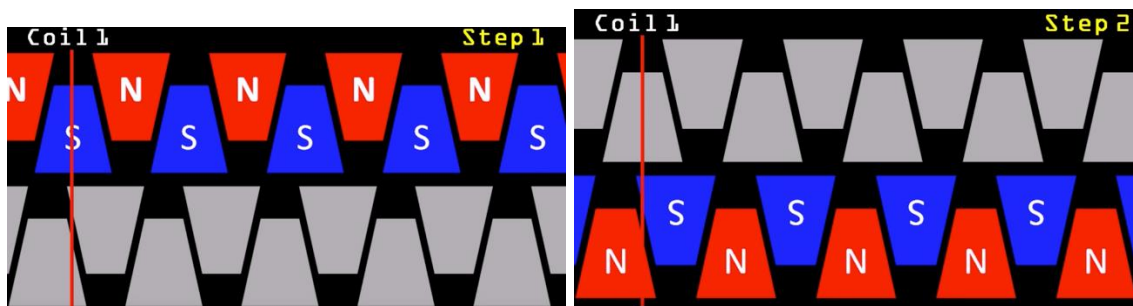
L'eix central, és a dir el que inicia el moviment, forma part d'un cilindre imantat amb 16 pols; 8 pols nord i 8 pols sud. Aquest cilindre va encaixat al que seria la carcassa exterior, la qual té 4 pisos de 8 dents com es pot veure en la foto. Tenint un total de 32 dents que formaran una volta del primer eix. Anomenarem pas cada dent moguda. Per tant perquè l'eix de sortida faci una volta caldran 4096 passos.



Fem passar electricitat a través de la peça que envolta el cilindre per tal de generar un camp magnètic. Canviant el sentit del corrent elèctric canviarà l'orientació dels pols com podem observar a la imatge que hi ha a continuació.



A conseqüència que els pols canviïn de posició aconseguim moure el motor. Si comencem amb el pol nord a la part superior, el cilindre central que també està imantat es mourà de manera que la seva franja de pol sud coincideix amb el pol nord de la peça que l'envolta. Per tant el que fem és canviar el sentit del corrent que passa per aquesta peça, de tal manera que ara tenim els pols sud a la part superior. Això provoca que el cilindre es mogui per tornar a tenir el seu pol sud amb el nord que ara està a la part inferior i per tant una mica més endavant. En la representació gràfica es veu el funcionament, on la línia recta vermella és el cilindre.



- **Numeric pad:** és un element d'entrada que serveix per a escriure un codi i que la Raspberry Pi el pugui llegir i interpretar. Per detectar quina tecla estàs prement, es posen a zero totes les columnes i després es va preguntant quina és la fila de la qual li arriba una entrada, seguidament escriu quina és la posició dins la matriu, és a dir la seva fila i columna. En el codi has de declarar la matriu per poder treballar amb files i columnes.

- **Char LCD (2x16):** Un altre component és una pantalla (char lcd 2x16). Simplement és una pantalla adaptada per a la Raspberry Pi, en la que pots veure valors de la programació, com en el nostre cas el codi pitjat en el numeric pad. La que s'utilitza en aquesta maqueta és el model que té dues files i setze columnes, tot i que aquest valor pot ser diferent.

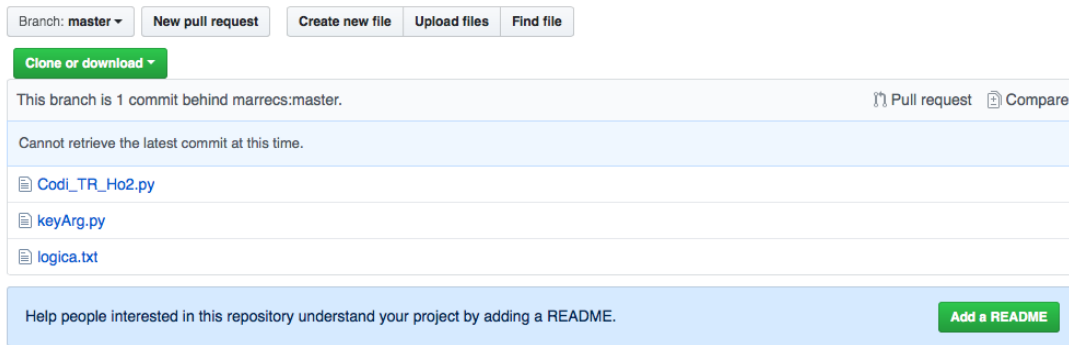


- **Càmera:** També utilitzo una càmera per a la Raspberry Pi. En el meu programa el que aquesta fa és fer fotos i enviar-les a un directori, en una carpeta de la Raspberry Pi.



Per programar a la Raspberry Pi a més del Python he utilitzat una plataforma anomenada Github. És una plataforma de desenvolupament col·laboratiu per a emmagatzemar projectes de programació. On crees els teus repositoris Git; una còpia del codi local. Els repositoris són versions del teu projecte que es troben a internet, en aquest cas a la teva conta del Github. En fer variacions, es crea una còpia de l'original, així sense haver de modificar-lo. Per tant dins el repositori trobes més d'un programa. Aquesta plataforma pot ser gratuïta, amb la condició que tothom té accés als teus programes. Això sí, al fer alguna modificació, reps un avís i tu tries si vols fer el canvi. Per això les grans empreses paguen per tenir en privat els seus projectes i que la gent no els pugui copiar. Només els programadors treballant-hi tenen accés.

A la foto es pot veure un repositori amb els tres programes.



El segon tipus de programació utilitzat és bastant antònim a la Raspberry Pi. El Lego Mindstorm és una línia de robòtica creada per l'empresa Lego. Els robots utilitzats estan fets amb peces d'aquesta empresa, un cervell i la programació és a través de l'app "ev3 Programmer". És un programa de blocs en comptes de codi, el que el fa molt més visual i senzill. A més no és una placa sinó un brick intel·ligent que funciona com a cervell del robot, el qual controla els motors i els sensors que li connectis. Té connexió sense fil de Wi-Fi i Bluetooth. Aquest robot consta de tres ports de sortida identificats amb lletres i quatre ports d'entrada numerats. Els sensors són components d'entrada i en canvi els motors de sortida.



Cervell del Lego
Mindstorm

El sensor d'ultrasò el que fa és detectar la distància de l'element que tingui a davant. Ho fa generant ones de so d'alta freqüència i llegint el retràs dels seus ecos per a detectar i mesurar la distància. També pot enviar ones de so individuals per a treballar com a sonar o per escoltar una ona de so que desencadena l'inici d'un programa. Mesura distàncies entre 1 i 250 cm amb una precisió de +/- 1cm.



També s'utilitza un sensor de tacte que detecta la pressió, per exemple pitjar o deixar anar un botó, per tant ens diu si el botó està premut o no.

De sortida en la maqueta trobem servomotors, ens permet controlar amb precisió la posició i velocitat amb 1 grau d'exactitud.

A banda de la programació he utilitzat altres components robòtics com la impressora 3D, una tecnologia molt present en el futur que no deixa d'avançar. És una màquina capaç d'agafar un disseny de l'ordinador fet amb software de modelatge 3D i imprimir-ne una figura en volum. Els arxius d'aquests programes contenen instruccions precises sobre les coordenades que la màquina ha de seguir per a crear l'objecte. La producció és per capes, la broqueta és escalfada de manera que injecta plàstic fos, que un cop dipositat s'asseca immediatament, donant lloc a la nova peça.

Per fer els diferents dissenys he utilitzat el Tinkercad, un software gratuït online per crear dissenys 3D. Destaca per ser molt intuïtiu i fàcil d'utilitzar, el que també fa que les funcions més avançades i complexes no estiguin disponibles.

També he fet ús d'un programa per a fer codis Qr, jo he utilitzat el code monkey. El seu ús és tan senzill com afegir el enllaç de la pàgina a la qual voldràs que dirigeixi el codi QR. A més el pots dissenyar al teu gust.

El Google Maps també ha estat un programa usat en el meu projecte, l'he fet servir per crear la geolocalització de les diferents màquines.

Per últim he utilitzat l'app inventor, un entorn de desenvolupament de Software creat per Google Labs per l'elaboració d'aplicacions destinades al sistema operatiu d'Android.

LA MÀQUINA EXPENEDORA

La idea innovadora que jo proposo a aquest problema és una màquina expenedora d'aigua. En la qual la gent tingués accés a aigua gratuïta i de la xarxa filtrada. Aquesta té moltes més propietats per a la salut. En ella també podries comprar ampolles de vidre i ampolles termo, a escollir, així tenint preus per a tothom. Sempre que no portessis la teva, que també podria ser utilitzada. La màquina també constaria d'un sistema de retorn de les ampolles comprades, en el cas que el ciutadà ja no la volgués. D'aquesta manera encara estariem allargant més la vida útil d'aquella ampolla. Aquest cicle també tindria un procés de rentatge de totes les ampolles retornades. En realitzar aquesta funció et tronarien part del preu inicial pagat, així motivant a la gent de no llençar-les. La màquina funcionaria amb una app, que generaria un codi cada vegada que compressis o retornessis una ampolla per tal de controlar que no es manipulés el sistema amb altres ampolles. L'altre avantatge de l'aplicació és que facilitaria el pagament. Actualment portem més sovint a sobre el mòbil que les monedes. Hi hauria l'opció que entitats interessades en el projecte i ajudar el medi ambient podrien participar i proporcionar descomptes o avantatges a tot aquell qui fes ús de la màquina. En el seu interior hi hauria una càmera per controlar l'estat intern i evitar viatges innecessaris del personal de manteniment.

La idea inicial seria que els ajuntaments les financessin, ja que és una millora per als seus ciutadans.

Un cop tenia la idea estructurada vaig decidir que volia fer una maqueta per ensenyar de manera més visual la idea. La maqueta és funcional, per tant no és igual a la real.

REALITZACIÓ DE LA MAQUETA

CONSTRUCCIÓ DE L'ESTRUCTURA

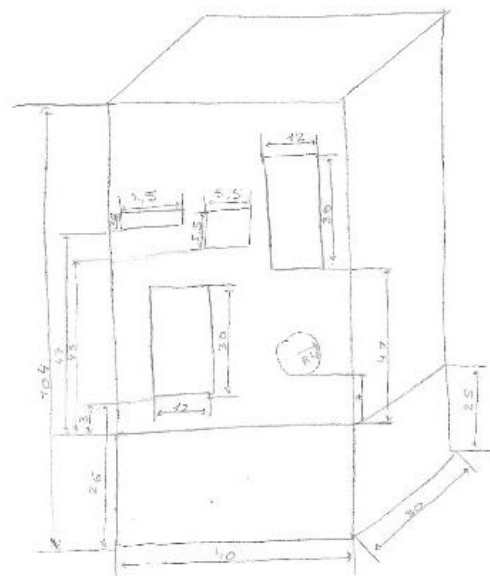
MATERIALS I EINES

- Trepant
- Tornavís
- Peu de rei
- Cinta mètrica
- Claus
- 2 metacrilats 800x300
- 2 contraplacats de pi 800x360
- Caixa de fusta 250x400x300
- Tablex perforat 350x300
- Tablex perforat
- 4 llistons d'avet 20x20x800
- 2 fronisses
- Imant
- Escaires
- Perfils en forma de 'L'
- Ampolla de vidre

PROCÉS DE CONSTRUCCIÓ

El primer pas a fer és dissenyar i escollir els materials de la maqueta. És important fer un bon croquis per tenir clares les mides i l'estructura.

Jo vaig decidir construir els dos laterals amb metacrilat per tal que el seu interior quedés a la vista. Així podria ensenyar tot el mecanisme i com queda



estructurada per dins. La cara posterior volia que fos una porta de fusta, d'aquesta manera sempre podré modificar l'interior o reparar qualsevol avaria del mecanisme o de les connexions. La cara del davant també seria de fusta amb tots els forats necessaris per a les funcions de la màquina. Finalment vaig decidir que la base volia que fos una caixa de fusta, per tal d'assegurar l'estabilitat. Altrament veia molt complicada la tasca d'unir les quatre cares a una base de fusta. Per incrementar més aquesta estabilitat i facilitar encara més la junta de les quatre cares, quatre llistons de fusta surten de les quatre cantonades de la caixa fins a gairebé la part més alta de la màquina. Per l'interior, volia posar un tablex perforat de la mateixa mida que la base però a l'alçada del forat per on surten les ampolles. També en necessitaria un altre a l'alçada del forat per omplir l'ampolla d'aigua, però aquest cop seria la meitat de l'amplada de la màquina i tota la seva profunditat. Això és degut al fet que l'altra meitat ha de quedar lliure per la caiguda de les ampolles. Així em serviren de bases per posar els mecanismes i de base per les ampolles caigudes i l'ampolla que es volgués reomplir. Sobre aquesta base també posaria una cinta transportadora per on sortirien les ampolles.

També caldrà decidir quin material protegirà la caiguda de les ampolles de vidre en el retorn, per les seves característiques el més recomanat seria un suro.

També vaig decidir que el mecanisme el faria amb una cinta transportadora en comptes del sistema real que podem observar en el segon plànol.

Abans d'anar a comprar els materials, per tenir ben clares les mides i que la forma fos funcional vaig fer un disseny 3D amb el Tinkercad. Amb aquest es poden posar les mides de cada peça i es veu millor que amb un dibuix si la idea es pot dur a terme.

Un cop amb tots els materials toca començar a construir-la. El primer pas és clavar els quatre llistons a les cantonades de la caixa de fusta.



S'ha d'anar amb compte a l'hora de collar els claus, ja que pot passar que no quedi agafat el llistó amb el clau. Per això s'ha de pressionar bé el llistó contra la paret de la caixa per superar la força que fa el trepant amb el clau per separar-lo i que no quedi ben collat.

Un cop ben fix els llistons s'ha de col·locar el metacrilat. Només en posarem un de moment, ja que si tinguéssim les quatre cares dificultaria la tasca de posar les rapises interiors.

Cal col·locar el metacrilat a la seva posició per marcar on aniran els forats dels claus. Un cop marcats s'ha de posar la peça en una superfície plana amb una fusta a sota. Seguidament amb un trepant i una broca podem procedir a fer els forats de manera molt meticulosa per a no esquerdar la superfície. Un cop fets els forats és recomanable fer el forat al llistó d'on anirà el clau. Per últim ja podem clavar el metacrilat.



A l'hora de clavar el clau, com ja tenim fet el forat és important utilitzar el tornavís i no el trepant, ja que sinó el metacrilat s'esquerda. A causa de l'esquerda vaig haver de repetir el pas.



Abans de collar el contraplacat s'han de marcar i tallar els forats desitjats. En aquest cas he dibuixat un rectangle a la part inferior per posar l'ampolla i omplir-la, un rectangle una mica més amunt per on sortiria l'ampolla comprada i un cercle per retornar les ampolles.



Un cop fets els talls ja es pot collar el contraplacat als llistons.



Seguidament s'ha de posar la porta. D'entrada cal collar bé les frontisses, primer als llistons i després marcant a la porta amb un llapis on anirà la part corresponent de la peça. Un cop feta la marca podem treure la meitat de la frontissa que anirà a la porta i treballar només al contraplacat.



Un cop posades les quatre parts de les frontisses ja podem encaixar la porta.



D'altra banda també cal pensar en un sistema perquè la porta quedi ben tancada. Jo he triat un imant. Al llistó he caragolat l'imant i a la porta he enganxat el metall. Cal anar amb compte que al col·locar l'imant segueixi tancant bé la porta. Per això ens hem de fixar que quedi ben exacte a la cantonada del llistó.



El següent pas és col·locar la base que farà de suport per al mecanisme a l'alçada del forat per on sortiran les ampolles. Per tenir més espai a l'hora de treballar traurem la porta. A continuació s'ha de marcar als quatre llistons on

aniran els escaires per suportar la plataforma. Un cop fetes les marques collarem els escaires fins que es pugui col·locar el tablex.



Cal anar en compte que aquesta alçada no coincideixi amb cap nus dels llistons ja que aquestes zones consten més de foradar.



A continuació farem el mateix per al prestatge inferior. Tornarem a marcar la mida pertinent i clavarem els escaires fins que puguem recolzar el tablex al damunt.

Ara que ja no haurem de treballar més en l'interior podem tornar a posar la porta. També podem tancar les quatre cares i posar el metacrilat. Tornarem a seguir els mateixos passos que amb el primer. Cal vigilar que quedi a la mateixa posició que l'altre.

Així hem acabat el que seria l'estructura de la maqueta.

PROGRAMACIÓ DE LA MAQUETA

MATERIAL NECESSARI

- Soldador
- Voltímetre
- Raspberry Pi 3
- Monitor
- Ratolí
- Teclat
- Cables
- Breadboard
- Motor pas a pas
- Càmera Raspberry Pi3
- Numeric pad
- Resistències
- Char lcd
- Cable per a la breadboard
- Cable hdmi
- Cable de corrent per a la Raspberry Pi
- Impressora 3D
- 1 brick NXT
- Cables connectors mindstorms
- 2 motors grans
- 1 sensor d'ultrasò
- 1 sensor de tacte

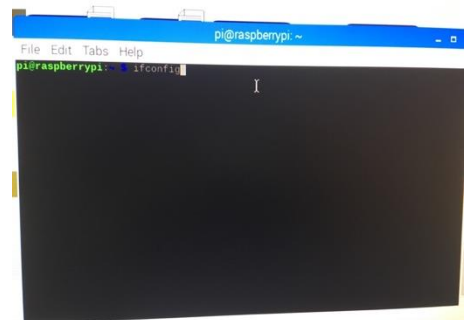
PROCÉS DE PROGRAMACIÓ

Jo sempre havia programat amb Lego Mindstorm, i tot i que ja tens la dinàmica de la programació, t'has d'anar adaptant al nou llenguatge, el que ralentitza molts passos. Com la placa Raspberry Pi té un nombre de pins limitats a connectar i necessitava connectar més motors i elements dels permessos vaig decidir que en comptes de comprar un altra placa inclouria també en la maqueta elements programats amb el NXT del Lego Mindstorms.

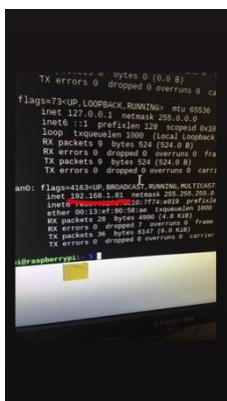


La Raspberry Pi és com un ordinador, és per això que el primer que s'ha de fer és connectar la placa a el corrent, a un monitor. Per tant, a les entrades USB hem de connectar el teclat i el ratolí. Un cop a la pantalla principal, s'ha de configurar la xarxa Wi-Fi.

Un cop configurada, jo volia poder controlar-la des del meu portàtil. Per fer-ho s'ha de cercar l'adreça IP de la Raspberry Pi. Aquesta es troba a la consola de comandaments de la Raspberry. En allà s'ha d'escriure "ifconfig", de seguida surt tota la informació de la placa, amb la IP inclosa.



El comando des d'on podem veure la IP.



Podem veure la IP subratllada

Un cop tens la IP, des del portàtil s'ha d'instal·lar el Vnc viewer. En aquest programa només has de posar l'adreça IP, l'usuari i la contrasenya i ja t'apareix la pantalla principal de la Raspberry Pi. Tot i fer de pantalla de la Raspberry Pi, aquesta es pot minimitzar i seguir utilitzant el teu ordinador. Tot i que com la Raspberry Pi és un ordinador també té internet.

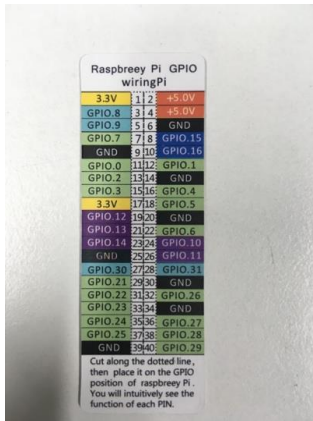


Ara ja podem programar des del portàtil amb el Python.

Abans de començar a programar s'ha de tenir molt clar quins són els components necessaris per a cada moviment, i les ordres que s'hauran d'enviar a la placa per dur-lo a terme. Primer s'ha d'estructurar, com que estem acostumats al llenguatge humà, en canvi el llenguatge d'una màquina s'ha d'especificar tot, fins al que sembla evident. Per tant el primer pas no és agafar l'ordinador, sinó un paper i un llapis per anotar les instruccions que més endavant hauràs de programar. D'aquesta manera després t'estalviaràs molts errors i temps.

Segons la meva manera de fer, és molt recomanable programar cada component per separat i al final ajuntar-los tots en un sol programa.

Per tal de tenir més espai per connectar els pins he utilitzat una breadboard. Per tant he utilitzat un cable especial per connectar la breadboard amb la placa. S'ha d'utilitzar una guia per veure a quin pin correspondrà cada fila de la breadboard i així connectar del dret el cable. A més ho he posat tot a sobre d'una plataforma per tal que quedi ferm.

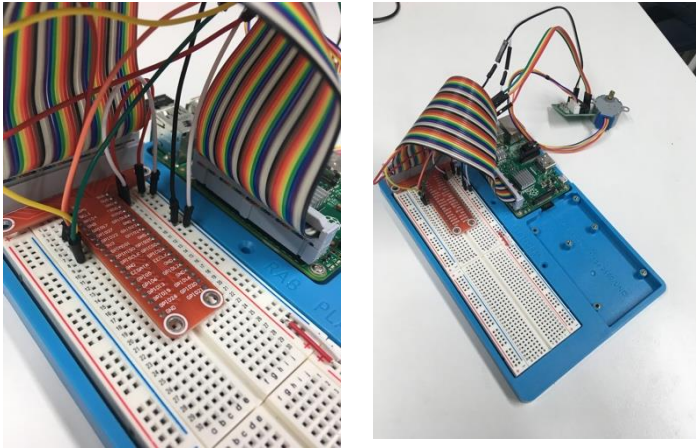


Aquest cable s'ha de connectar; l'extrem de plàstic gris ha d'anar enganxat als 40 pins de la placa. D'aquesta manera els pins quedaran connectats a la breadboard, cadascun en el forat corresponent segons la nova col·locació, indicada amb la placa vermella del cable.

Tot i que la guia ens mostrava la correspondència de pins i per tant com havia d'anar col·locat el cable, mai està de menys comprovar-ho. I per tant per saber si aquest cable estava del dret o del revés vaig utilitzar un voltímetre. Vaig fer una connexió qualsevol de prova i connectar el voltímetre. De tal manera que si estava ben connectat i per tant el circuit tancat sonaria un soroll. Com aquesta segona manera és més ràpida cada vegada que desconnectava i tornava a connectar el cable utilitzava aquest segon sistema.



Quan la placa ja està sobre el suport i amb la breadboard preparada ja podem muntar el circuit del primer component.



Doncs jo vaig voler començar a programar el motor pas a pas. És un motor de gran precisió el que provoca també que sigui bastant lent. Primer, s'ha de connectar el motor a la placa. Cal tenir compte els pins que utilitzem i anotar, ja que els haurem de declarar en el programa. Si els pins no són els correctes, el programa no funcionarà.

Cal buscar el funcionament del motor per poder entendre'l i escriure el programa. Cal assegurar-se que tens descarregades les llibreries declarades. Un cop fet el programa s'ha de calcular els graus que es vol fer girar l'hèlix i fer el canvi necessari en el programa. Jo he canviat el programa per tal que en comptes d'haver de posar els passos que vols que fagi el motor puguis posar els graus. Això ho he fet amb una fórmula tan senzilla com multiplicar els graus declarats multiplicar-los per 512 passos que són una volta i es divideix entre 360.

El programa és el següent:

```

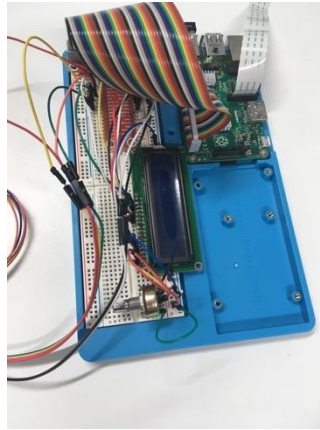
stepper_motor_definitiu.py - /home/pi/Desktop/stepper_motor_definitiu.py (2.7.13)
File Edit Format Run Options Window Help
|
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
control_pins = [4,27,10,9]
numero = 360
for pin in control_pins:
    GPIO.setup(pin, GPIO.OUT)
    GPIO.output(pin, 0)
halfstep_seq = [
    [1,0,0,0],
    [1,1,0,0],
    [0,1,0,0],
    [0,1,1,0],
    [0,0,1,0],
    [0,0,1,1],
    [0,0,0,1],
    [1,0,0,1]
]
for i in range(numero*512/360): #512 una volta
    for halfstep in range(8):
        for pin in range(4):
            GPIO.output(control_pins[pin], halfstep_seq[halfstep][pin])
            time.sleep(0.001)
GPIO.cleanup()

```

El següent component a programar és la pantalla. Aquesta ve amb les potes metàl·liques per connectar la pantalla per separat. És per això que s'han de soldar. Quan soldem el que estem fent és escalfar l'estany perquè es fongui sobre dues parts també metàl·liques, les potes i la pantalla en aquest cas. En refredar-se queden unides les dues parts amb l'estany. S'ha d'anar amb compte de no soldar massa, ja que pot cremar la placa.



Ara ja podem connectar la pantalla a la breadboard, en els seus pins corresponents.



Hi ha un programa molt senzill que es diu “Hello world” que serveix per aprendre el funcionament d’aquest component. Aquest programa simplement escriu a la pantalla aquesta frase. I doncs això és el primer a fer quan tens tots els cables connectats. El següent programa que vaig provar de fer va ser un que ensenyés a la pantalla els passos que anava avançant el motor. També era una manera d’estudiar millor el seu funcionament i començar a connectar dos components.

```
Programma_pantalla_lcd.py - /home/pi/Desktop/Programma_pantalla_lcd.py (2.7.13)
File Edit Format Run Options Window Help
import time
import Adafruit_CharLCD as LCD
import sys
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM)

control_pins = [4,27,10,9]

numero = 90

## Raspberry Pi pin configuration de la pantalla:
lcd_rs = 25
lcd_en = 24
lcd_d4 = 23
lcd_d5 = 17
lcd_d6 = 18
lcd_d7 = 22
lcd_backlight = 2

##definim la mida de les columnes i files 16x2
lcd_columns = 16
lcd_rows = 2

lcd = LCD.Adafruit_CharLCD(lcd_rs, lcd_en, lcd_d4, lcd_d5, lcd_d6,
                           lcd_d7, lcd_columns, lcd_rows, lcd_backlight)

lcd.message('Hello\nworld!')
```

```

Programma_pantalla_lcd.py - /home/pi/Desktop/Programma_pantalla_lcd.py (2.7.13)
File Edit Format Run Options Window Help

##esperar 5 segons
time.sleep(5.0)
lcd.clear()

for pin in control_pins:
    GPIO.setup(pin, GPIO.OUT)
    GPIO.output(pin,0)

halfstep_seq = [
    [1,0,0,0],
    [1,1,0,0],
    [0,1,0,0],
    [0,1,1,0],
    [0,0,1,0],
    [0,0,1,1],
    [0,0,0,1],
    [1,0,0,1]
]

for i in range(512*numero/360):
    for halfstep in range(8):
        for pin in range(4):
            GPIO.output(control_pins[pin], halfstep_seq[halfstep][pin])
            lcd.clear()
            lcd.message('Pin:%d Ha:%d I:%d' %(pin, halfstep,i))
            time.sleep(0.10101)

GPIO.cleanup()

```

La càmera era el següent pas. Aquest programa era l'únic que ja sabia fer. Ja havia utilitzat una càmera per a la Raspberry Pi en un projecte anterior. Així que ja vaig anar directe al programa definitiu. Aquest programa que s'està executant va fent fotos amb l'espai de temps declarat i les guarda a una carpeta. És important intentar fer el programa per tal que guardi la foto amb la data i l'hora. Aquest incís és important perquè si no per defecte guarda les fotos amb el mateix nom i per tant cada vegada que fa una foto i la guarda reemplaça l'anterior. Jo he utilitzat el sistema de la data i l'hora, ja que en el que seria la màquina real crec que seria interessant. Però és clar que hi ha altres alternatives per a solucionar aquest problema.

```

camara.py - /home/pi/Desktop/camara.py (2.7.13)
File Edit Format Run Options Window Help
|
from picamera import PiCamera
from time import sleep
import datetime

##create object for PiCamera class
camera = PiCamera()
#DateTime = str(datetime.datetime.now())

#DateTime.Now.ToString("yyyy-dd-M--HH-mm-ss");
date_string = datetime.datetime.now().strftime("%Y-%m-%d-%H:%M")
#camera.start_preview()
#s#leep(10)
print('/home/pi/Desktop/image'+date_string+'.jpg')
camera.capture('/home/pi/Desktop/image'+date_string+'.jpg')
camera.stop_preview()
sleep(3600)

```

Cal afegir que la posició del cable és important per al funcionament de la càmera. Aquest cable és pla i té una ranura indicada en la placa on s'ha de connectar. Aquest té una marca blava al final, aquesta ha de quedar mirant a les ranures USB, tal com es mostra a la foto.



La definició de la càmera no és excessivament bona, al cap i a la fi per la maqueta ja és suficient.

L'últim programa dels components és el del numeric pad. El programa utilitza matrius per declarar les files i columnes. El primer programa escrivia en la pantalla de l'ordinador els números que premies.

```

numeric pad.py - /home/pi/Desktop/numeric pad.py (2.7.13)
File Edit Format Run Options Window Help
|
import sys
import RPi.GPIO as GPIO
import time

#GPIO.setmode(GPIO.BOARD)
GPIO.setmode(GPIO.BCM)
#GPIO.setwarnings(False)

MATRIX = [ [1,2,3,'A'],
            [4,5,6,'B'],
            [7,8,9,'C'],
            ['*',0,'#','D']]

#ROW = [7,11,13,15]
#COL = [12,16,18,22]

ROW = [21,20,16,12] #negres
COL = [26,19,13,6] #blancs #18,23,24,25]

print ("Pas1")

for j in range(4):
    #print [j]
    #print COL[j]

    GPIO.setup(COL[j], GPIO.OUT)
    GPIO.output(COL[j],1)

for i in range(4):
    GPIO.setup(ROW[i], GPIO.IN, pull_up_down = GPIO.PUD_UP)
    #print [i]
    #print ROW[i]

try:
    while(True):
        for j in range(4):
            GPIO.output(COL[j],0)

            for i in range(4):

                if GPIO.input(ROW[i]) == 0:
                    print (MATRIX[i] [j])
                    time.sleep(0.2)

                    while(GPIO.input(ROW[i]) == 0):
                        pass

            GPIO.output(COL[j],1)

except KeyboardInterrupt:
    GPIO.cleanup()

```

Però el programa definitiu escriu aquests números, que a la pràctica seria un codi, a la pantalla de la màquina (char lcd).

```

*char_lcd_num_pad_2.py - /home/pi/Desktop/char_lcd_num_pad_2.py (2.7.13)*
File Edit Format Run Options Window Help
import time

import Adafruit_CharLCD as LCD
import sys
import RPi.GPIO as GPIO

#GPIO.setmode(GPIO.BOARD)
GPIO.setmode(GPIO.BCM)
#GPIO.setwarnings(False)

buzzer=5

GPIO.setup(buzzer, GPIO.OUT)

# Raspberry Pi pin configuration:
lcd_rs      = 25 # Note this might need to be changed to 21 for older revisio
lcd_en      = 24
lcd_d4      = 23
lcd_d5      = 17
lcd_d6      = 18
lcd_d7      = 22
lcd_backlight = 4

# Define LCD column and row size for 16x2 LCD.
lcd_columns = 16
lcd_rows    = 2

# Initialize the LCD using the pins above.
lcd = LCD.Adafruit_CharLCD(lcd_rs, lcd_en, lcd_d4, lcd_d5, lcd_d6, lcd_d7,
                           lcd_columns, lcd_rows, lcd_backlight)

# Print a two line message
lcd.message('Hello\nworld!')

# Wait 5 seconds
time.sleep(5.0)

# Demo showing the cursor.
lcd.clear()
lcd.show_cursor(True)
lcd.message('Show cursor')

time.sleep(5.0)

# Demo showing the blinking cursor.
lcd.clear()
lcd.blink(True)
lcd.message('Blink cursor')

time.sleep(5.0)

# Stop blinking and showing cursor.
lcd.show_cursor(False)
lcd.blink(False)

# Demo scrolling message right/left.
lcd.clear()
message = 'Scroll'

```

```

lcd.message(message)
for i in range(lcd_columns-len(message)):
    time.sleep(0.5)
    lcd.move_right()
for i in range(lcd_columns-len(message)):
    time.sleep(0.5)
    lcd.move_left()

# Demo turning backlight off and on.
lcd.clear()
lcd.message('Flash backlight\nin 5 seconds...')
time.sleep(5.0)
# Turn backlight off.
lcd.set_backlight(0)
time.sleep(2.0)
# Change message.
lcd.clear()
lcd.message('Goodbye!')
# Turn backlight on.
lcd.set_backlight(1)

MATRIX = [ [1,2,3,'A'],
            [4,5,6,'B'],
            [7,8,9,'C'],
            ['*',0,'#','D']]

ROW = [21,20,16,12] #negres
COL = [26,19,13,6] #blancs #18,23,24,25

print ("Pas1")

for j in range(4):
    #print [j]
    #print COL[j]

    GPIO.setup(COL[j], GPIO.OUT)
    GPIO.output(COL[j],1)

for i in range(4):
    GPIO.setup(ROW[i], GPIO.IN, pull_up_down = GPIO.PUD_UP)
    #print [i]
    #print ROW[i]

try:
    while(True):
        for j in range(4):
            GPIO.output(COL[j],0)
            #print [j]
            #print "pas2"
            #print (MATRIX[i] [j])

        for i in range(4):
            #print (MATRIX[i] [j])
            #time.sleep(0.82)

```



```

#print i
#print "pas3"
if GPIO.input(ROW[i]) == 0:
    print (MATRIX[i] [j])
    lcd.clear()
    lcd.message('Has premut ' + str(MATRIX[i] [j]))
    for temps in range(MATRIX[i] [j]):
        GPIO.output(buzzer,GPIO.HIGH)
        print('Beep')
        time.sleep(0.5)
        GPIO.output(buzzer,GPIO.LOW)
        print('No Beep')
        time.sleep(0.5)
    time.sleep(0.2)

    while(GPIO.input(ROW[i]) == 0):
        pass

        GPIO.output(COL[j],1)
except KeyboardInterrupt:
    GPIO.cleanup()

```

Un cop tenim tots els components programats cal fer la lògica del programa a paper un altre cop abans de passar-la a l'ordinador. Tenint molt presents tot allò que vols que faci la màquina. S'ha d'escriure amb tot detall que haurà de fer la màquina en cada moment per estructurar el programa general, ja que tindrà molt mini programes al seu interior.

Un cop la lògica del programa està ben clara ajuda molt fer blocs de les diferents funcions que farà el programa més gran. Quan fem un bloc el que estem fent és subdividir el programa, els definim com a funcions. Aquestes funcions en el meu cas són els programes de cada component. Així després el podrem cridar amb aquell nom i no haurem d'escriure tot el programa sencer cada vegada que vulguem utilitzar aquell segment de programa. Per tant al principi definirem tots aquests programes, per a després només haver-los de cridar.

A continuació es pot veure el programa sencer:

```

tr.py - /home/pi/Desktop/tr.py (2.7.13)
File Edit Format Run Options Window Help
#Treball Recerca H2O
#13-12-2018

#Importem lliberies
import time
import Adafruit_CharLCD as LCD
import sys
import RPi.GPIO as GPIO
from picamera import PiCamera
from time import sleep
import datetime

camera = PiCamera()

date_string = datetime.datetime.now().strftime("%Y-%m-%d-%H:%M")
date_string = datetime.datetime.now().strftime("%Y-%m-%d-%H:%M")

#GPIO.setmode(GPIO.BOARD)
GPIO.setmode(GPIO.BCM)
#GPIO.setwarnings(False)

buzzer=5

GPIO.setup(buzzer, GPIO.OUT)
control_pins = [4, 27, 10, 9]

for pin in control_pins:
    GPIO.setup(pin, GPIO.OUT)
    GPIO.output(pin, 0)

halfstep_seq = [
    [1,0,0,0],
    [1,1,0,0],
    [0,1,0,0],
    [0,1,1,0],
    [0,0,1,0],
    [0,0,1,1],
    [0,0,0,1],
    [1,0,0,1]
]

# Raspberry Pi pin configuration:
lcd_rs      = 25 # Note this might need to be changed to 21 for older revisio
lcd_en      = 24
lcd_d4      = 23
lcd_d5      = 17
lcd_d6      = 18
lcd_d7      = 22
lcd_backlight = 4

# Define LCD column and row size for 16x2 LCD.
lcd_columns = 16
lcd_rows    = 2

# Initialize the LCD using the pins above.
lcd = LCD.Adafruit_CharLCD(lcd_rs, lcd_en, lcd_d4, lcd_d5, lcd_d6, lcd_d7,
                           lcd_columns, lcd_rows, lcd_backlight)

```

```

-----, -----, -----,
MATRIX = [ [1,2,3,'A'],
           [4,5,6,'B'],
           [7,8,9,'C'],
           ['*',0,'#','D']]

ROW = [21,20,16,12] #negres
COL = [26,19,13,6] #blancs #18,23,24,25]
numeroentrat=''

def Funcmessage (textmessage):
    # Print a two line message
    lcd.clear()
    lcd.message(textmessage)

def Funcshowcursor ():
    # Demo showing the cursor.
    lcd.clear()
    lcd.show_cursor(True)
    lcd.message('Show cursor')

def Funcblink (textBlink):
    # Demo showing the blinking cursor.
    lcd.clear()
    lcd.blink(True)
    lcd.message(textBlink)

def Funcstopblink ():
    # Stop blinking and showing cursor.
    lcd.show_cursor(False)
    lcd.blink(False)

def FuncScroll(textScroll):
    # Demo scrolling message right/left.
    lcd.clear()
    message = textScroll
    lcd.message(message)
    for i in range(lcd_columns-len(message)):
        time.sleep(0.5)
        lcd.move_right()
    for i in range(lcd_columns-len(message)):
        time.sleep(0.5)
        lcd.move_left()

def FuncSteppermotor (numero ):
    for i in range(numero*512/360): #512 una volta
        for halfstep in range(8):
            for pin in range(4):
                GPIO.output(control_pins[pin], halfstep_seq[halfstep][pin])
            time.sleep(0.001)
    GPIO.cleanup()

```

```

def Funcnumpad ():
    for j in range(4):
        #print [j]
        #print COL[j]

        GPIO.setup(COL[j], GPIO.OUT)
        GPIO.output(COL[j],1)

    for i in range(4):
        GPIO.setup(ROW[i], GPIO.IN, pull_up_down = GPIO.PUD_UP)
    try:
        while(True):
            if Funccomprova (numeroentrat)== 1:
                if numeroentrat == '1122':
                    Funcmessage ('codi correcte');
                    time.sleep (1);
                    lcd.clear();
                    lcd.Steppermotor (360);
                    lcd.clear();

            for j in range(4):
                GPIO.output(COL[j],0)
                for i in range(4):
                    if GPIO.input(ROW[i]) == 0:
                        print (MATRIX[i] [j])
                        global numeroentrat
                        if (MATRIX[i] [j]) == 'C':
                            numeroentrat = ''
                            lcd.clear()

                    else :
                        numeroentrat = numeroentrat + str(MATRIX[i] [j])
                        Funcmessage ('Has premut ' + numeroentrat)
                        time.sleep(0.2)

                    while(GPIO.input(ROW[i]) == 0):
                        pass

                GPIO.output(COL[j],1)
    except KeyboardInterrupt:
        GPIO.cleanup()

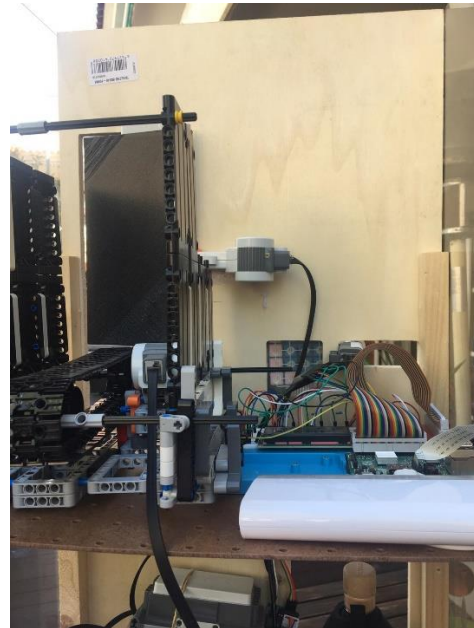
def Funccomprova (codi):
    ### si cert return 1
    if len (codi) == 4:
        if codi.isdigit ():
            return 1

def FuncCamara ():
    ##camera.start_preview()
    ##sleep(10)
    print('/home/pi/Desktop/image'+date_string+'.jpg')
    camera.capture('/home/pi/Desktop/image'+date_string+'.jpg')
    camera.stop_preview()
    sleep(3600)

##FuncScroll ('Hola');
Funcblink ('Escriu codi');
Funcnumpad();
FuncCamara();

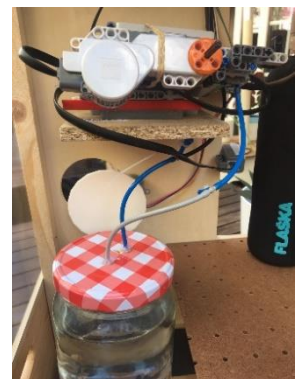
```

A la Raspberry Pi només tinc un motor connectat, i no hi ha prou pins disponibles com per a acoblar un altre. És per això que la resta de motors estan programats amb el Lego Mindstorms NXT. He decidit que el cervell del Mindstorm comandi la venda d'ampolles i la sortida d'aigua. Això és així perquè em falten dos motors que no puc suplir amb la Raspberry Pi, i aquesta funció n'utilitza dos i per tant han d'estar concatenats i iniciats per un mateix sensor, era més fàcil unir-los en un sol programa. Un motor mou la cinta transportadora i l'altre la porta. Això m'ha obligat a afegir un botó i que aquesta funció no estigui controlada pel panell amb números (numèric pad), ja que no és compatible amb Lego Mindstorm. El panell només controla la porta de retorn.

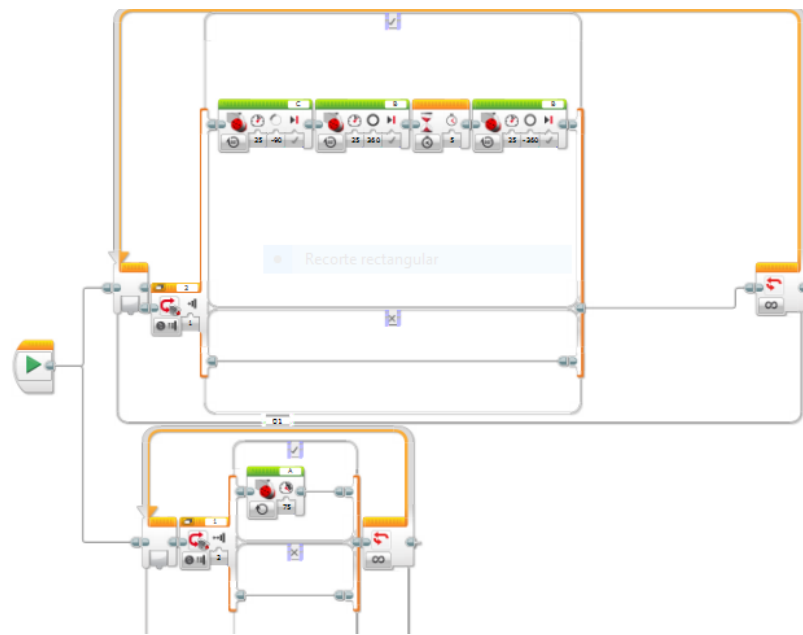


Aquesta és la cinta transportadora del sistema de recollida

La bomba ha estat un altre problema. En un principi la volia fer amb la Raspberry Pi, però no vaig trobar la manera. Per tant vaig decidir utilitzar el Lego Mindstorms que sí que en té un joc de pneumàtica amb el qual vaig crear la meua pròpia bomba. Aquesta bomba va connectada fins a un dipòsit amb aigua. D'aquesta manera es fa entrar aire per un conducte, l'aire comprimeix l'aigua de l'interior i fa que surti per un altre tub.



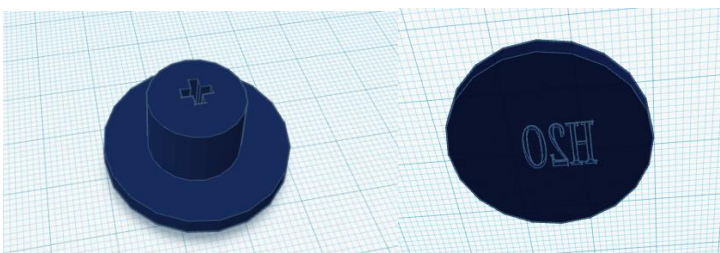
Per últim, aquí es pot veure el programa complet.



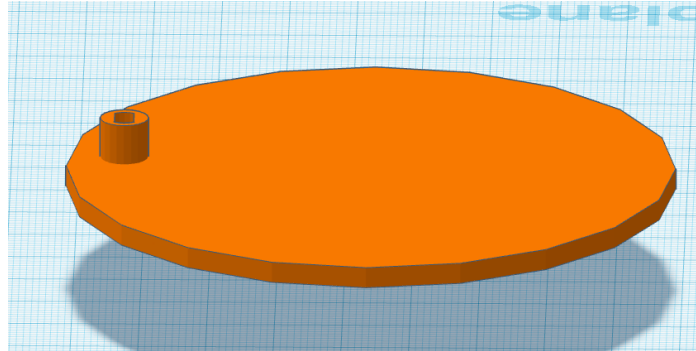
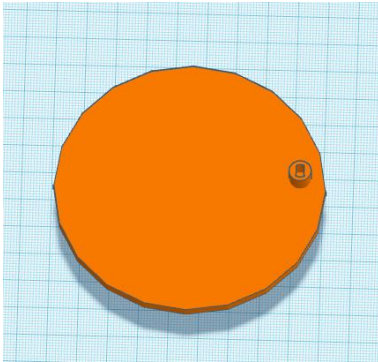
FINAL PROCÉS DE LA MAQUETA

Un cop muntada l'estructura i programat el mecanisme cal unir les dues parts. Per poder ajuntar-les i que la màquina sigui funcional s'han de fer alguns matisos. El primer pas és dissenyar la porta de retorn de les ampolles, que serà moguda per el motor pas a pas i per tant la Raspberry Pi. Això vol dir que caldrà fer una peça que s'adapti al seu eix i vagi fins a la porta. La manera més fàcil de dur-ho a terme és fent un disseny 3D d'una sola peça que surti un cilindre a mida de l'eix i continuï amb la forma de la porta. Cal tenir en compte que abans de començar a fer el disseny informàtic s'han de fer les mesures adients. S'haurà d'agafar el peu de rei per mesurar amb exactitud el diàmetre de l'eix de sortida del motor, el model utilitzat fa 5 mm. Seguidament hem de mesurar el forat on anirà la porta; en la nostra maqueta fa 8 cm i per assegurar que quedi ben tancat utilitzaré la mesura de 9 cm. A continuació cal fer el disseny amb el Tinkercad comprovant que les dues peces quedin ben unides.

Botó:



Porta:



Botó imprès
amb la
impresora 3D

També s'han imprès la porta i el logo amb el sistema d'impressió 3D.

Per últim he dissenyat una geolocalització a Google Maps i he creat el codi QR d'aquesta.



FUNCIONAMENT DE LA MAQUETA

La maqueta té un funcionament el més semblant a la realitat que ha estat possible. He aconseguit simular les opcions més bàsiques que tindria la real. L'aigua surt bombejada des d'un dipòsit que es troba l'interior de la maqueta.

Aquí trobem la primera diferència, ja que l'aigua no estaria extreta d'un dipòsit sinó de la xarxa. A més tampoc he pogut afegir el procés de filtratge o de refrigeració a la mostra. Per poder refrigerar la màquina es necessitarien elements incompatibles amb la Raspberry Pi.

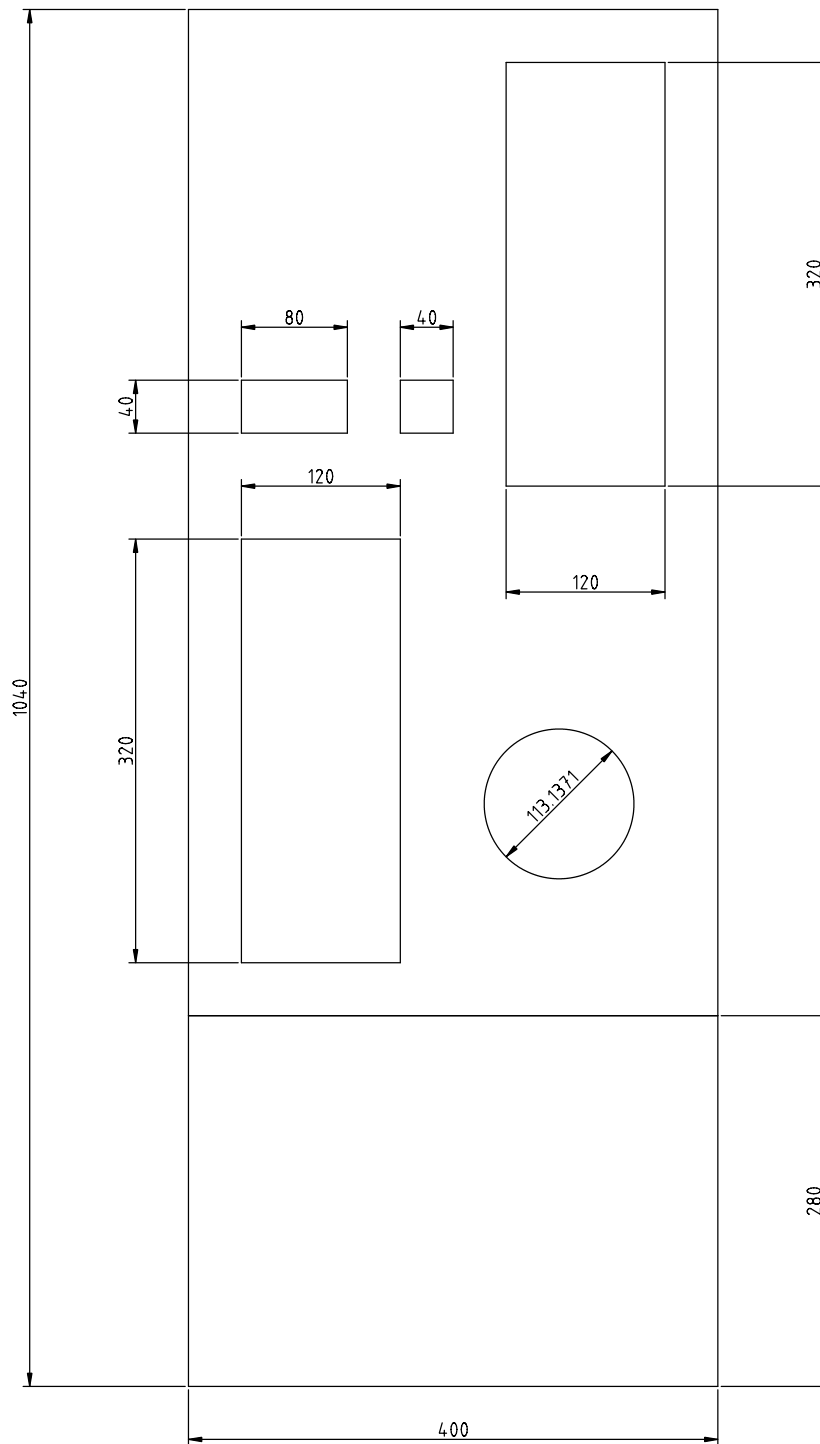
Tot i que la maqueta només emmagatzema 3 ampolles de vidre, la funció de comprar l'ampolla és bastant real. En la maqueta també has d'inserir un codi per poder-ne comprar una. A més el sistema és el mateix: les ampolles dipositades en una plataforma cauen a l'hèlix que aquesta, gira i en fa caure una.

El sistema de retorn tindria alguna mínima diferència a causa de les reduïdes dimensions de la maqueta. No podria tenir cap plataforma per evitar que les ampolles es trenquin i és per això que tindria un material que absorbeixi l'energia del cop.

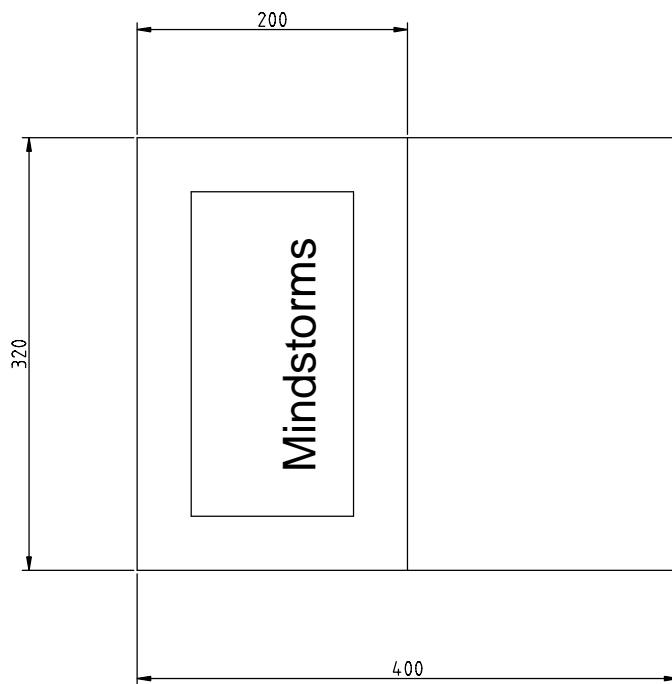
També he incorporat la càmera que faria un parell de fotos a cada jornada que envia a una conta tancada de Twitter. Des d'on el personal de manteniment podria comprovar l'estat interior i la disponibilitat de material.

Aquesta idea gràcies a diferents experts que m'han ajudat ha anat prenent forma. En molts aspectes m'han ajudat a veure opcions més visibles i m'han donat la seva opinió, des d'una perspectiva diferent i ja més enfocada a la realitat. Aquests col·laboradors han estat Flaska, Fill&drink. (Entevistes en els annexos dos).

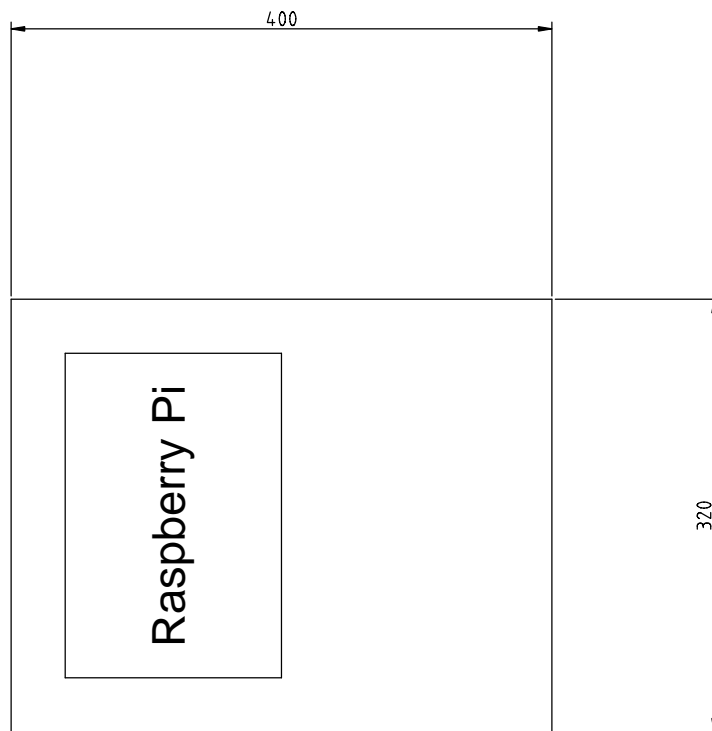
PLÀNOLS



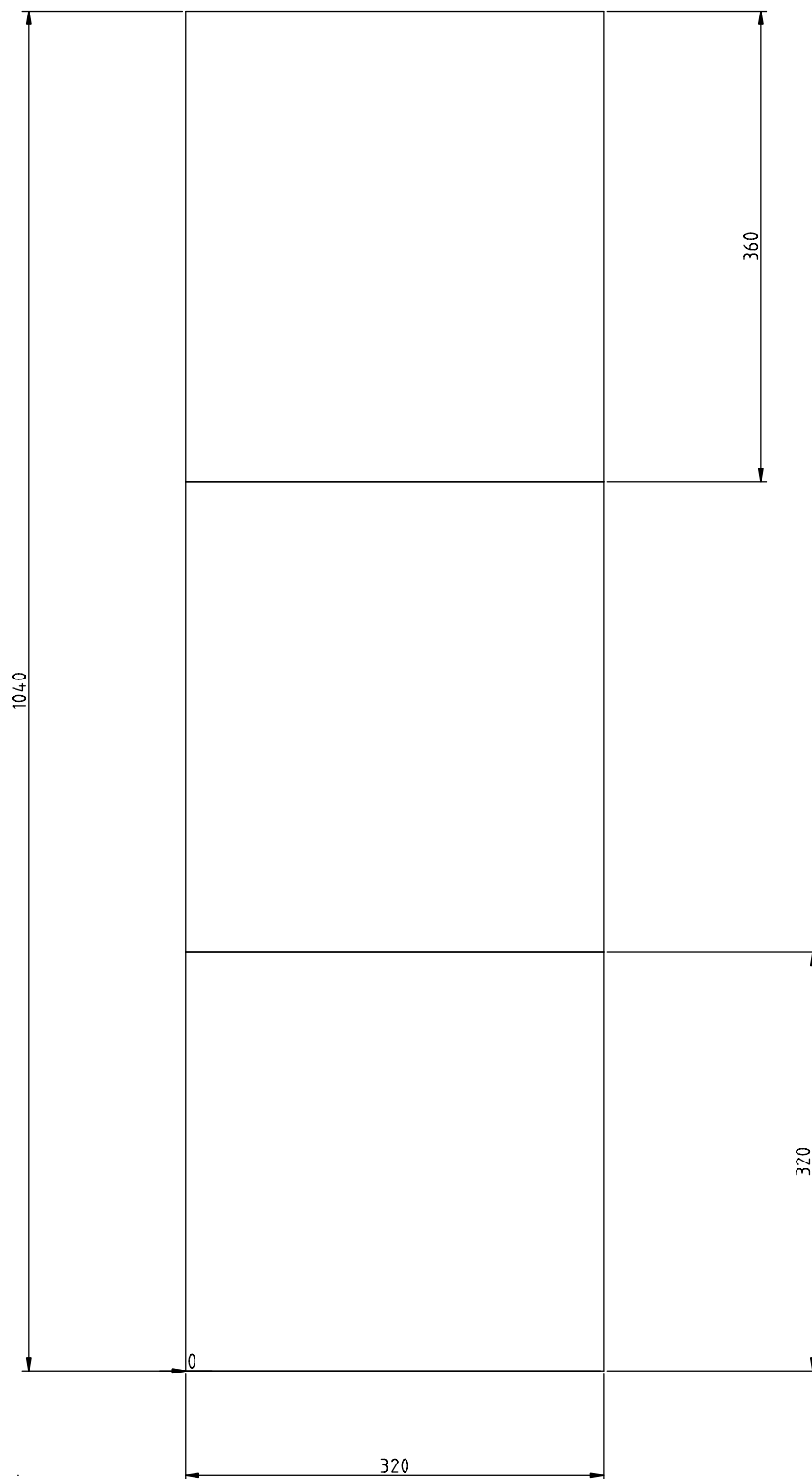
Alçat	INS Pere Ribot
Carla Claramunt	2n BAT B
20/12/2018	1:4



Planta 1	INS Pere Ribot
Carla Claramunt	2n BAT B
20/12/2018	1:4



Planta 2	INS Pere Ribot
Carla Claramunt	2n BAT B
20/12/2018	1:4



Alçat	INS Pere R
Carla Claramunt	2 BAT B
20/12/2018	1:4

RESULTATS

Després de tot el desenvolupament del projecte, finalment he aconseguit que la maqueta funcioni. Aquesta realitza totes les funcions bàsiques que faria la màquina real però amb petites variacions. Tot i que hi ha variacions són molt insignificants. A més es tracta de factors restringits pels sistemes de programació utilitzats.



CONCLUSIONS

Tal com he dit en l'apartat anterior, he assolit l'objectiu de fer una maqueta funcional, amb molt poca desviació de la idea inicial. La maqueta té totes les funcions que vaig plantejar en un inici.

Crec que també puc dir que he trobat una solució a un problema actual de la societat, com ho és la mala distribució de l'aigua a través del consum de les ampolles d'aigua. Una solució innovadora, que amb una mica més de desenvolupament en àmbits que jo no he treballat, com els materials de la màquina real, podria ser duta a terme. No obstant caldria acabar de calcular la viabilitat del projecte i fer un pressupost elaborat del qual seria la màquina. És una solució realista que milloraria la salut de la població. També seria de gran ajuda per al medi ambient, reduint o fins i tot gairebé eliminant les ampolles de plàstic. Una idea on òrgans com ajuntaments o altres organismes o grans empreses com Flaska estarien disposats a participar. No només això, sinó que l'Ajuntament de Vilassar de Mar proposava que presentés la idea als pressupostos participatius del poble, on es presenten projectes i el poble vota quins són els que prefereix finançar. Aquest pas per part de l'Ajuntament ja indica que la idea és realista i aplicable a l'actualitat. Això sí, s'hauria de continuar treballant en aquesta línia. De fet tots els professionals amb qui he contactat i compartit el meu projecte m'han encoratjat a continuar en aquesta recerca.

També és veritat que les empreses d'ampolles de plàstic estarien en contra. Però en tot projecte pel medi ambient hi ha un sector en contra perquè la proposta no vetlla pels seus privilegis o beneficis. Actualment existeix el problema de l'elevat preu de les ampolles de vidre. Un clar exemple són les ampolles de l'empresa Flaska; el preu d'aquestes volta els 30 euros. Un preu que molta gent no estaria disposada a pagar. És per això que fins que no es trobés una solució més eficaç, amb la finalitat de fer arribar el meu projecte a tothom, les ampolles no tindrien un gruix tan gran, fet que ja les abaratiria molt. Aquesta seria una solució temporal, ja que confio que l'avenç tecnològic trobarà una solució més econòmica, ja sigui a partir d'un nou material o millorant les característiques del vidre per tal d'abaratir-lo.

Si aquest projecte algun dia es traslladés a la realitat, jo situaria les màquines en espais públics, pel carrer i sobretot en llocs tancats com pot ser un centre comercial o el metro. Qualsevol lloc a l'abast de la ciutadania seria un bon lloc, ja que no requereix cap necessitat especial per a ser implantada. L'altre virtut d'aquesta màquina expendedora és que beneficia a tothom, no va dirigida a un públic concret. Al cap i a la fi tots necessitem l'aigua per viure.

Tot i el resultat satisfactori, si hagués de repetir el projecte, milloraria certes coses. Per exemple, intentaria fer un sistema de seguretat per evitar robatoris. També intentaria trobar una solució més ambiental a la refrigeració de l'aigua, ja que provoca una gran despesa d'energia. Per tant m'hagués agradat intentar posar plaques solars per tal de produir energia pròpia. Segurament l'estructura de la maqueta seria diferent també. Per exemple hauria tornat a intentar de fer el sistema real de compra d'ampolles. I és clar, l'estructura quedaria molt millor si l'hagués de tornar a fer, ja que m'he trobat amb petits problemes durant el procés. A més moltes de les eines les havia utilitzat molt poc abans, el que també provocava error d'execució.

En conclusió estic molt satisfeta de tot el que he après durant aquest procés i de l'experiència i coneixement que m'ha donat fer un treball així. També estic contenta amb el resultat final, a l'haver assolit els objectius marcats.

GLOSSARI

GPIO: sistema d'entrada i sortida de propòsit general, és a dir, consta d'una sèrie de pins o connexions que es poden utilitzar tan d'entrada com de sortida per a múltiples opcions.

SoC (System on Chip): processador potent que trobem en els ordinadors, depèn de xips o mòduls de memòria externs per a ser eficaços.

CPU (unitat central de processament): és el hardware de l'ordinador o altres dispositius programables, que interpreta les instruccions d'un programa informàtic mitjançant la realització de les operacions bàsiques aritmètiques, lògiques i d'entrada/sortida del sistema.

Memoria RAM (Random Access Memory): s'utilitza com a memòria de treball dels ordinadors per al sistema operatiu, els programes i la majoria del software. En la RAM es carreguen totes les instruccions que executen la unitat central de processament, a més de contenir les dades que manipulen els diferents programes.

BIBLIOGRAFIA - WEBGRAFIA

Aguakm0: <https://www.aguakmcero.com> 13/09/2018

Aquae: https://www.fundacionaquae.org/wiki-explora/40_vida/index.html
6/09/2018

Aquae: <https://www.fundacionaquae.org/blog/consejos-del-agua/10-beneficios-del-agua-para-nuestra-salud/> 6/10/2018

Arduino: <https://arduino.cl/que-es-arduino/> 1/12/2018

Circuit Basics: <http://www.circuitbasics.com/raspberry-pi-lcd-set-up-and-programming-in-python/>
1/10/2018

Deusto Formación: <https://www.deustoformacion.com/blog/programacion-diseno-web/que-es-para-que-sirve-github> 10/08/2018

Flaska: <https://www.flaska.es/el-vidrio/botellas-de-plastico> 19/11/2018

International Agency for Research on Cancer: <https://www.iarc.fr/> 1/09/2018

La Vanguardia: <https://www.lavanguardia.com/natural/20181024/452535415567/plastico-de-un-solo-uso.html> 27/06/2018

La Vanguardia: <https://www.lavanguardia.com/local/madrid/20181108/452781153448/batalla-agua-embotellada-gratis-madrid-restaurantes.html> 8/11/2018

Medium: https://medium.com/@Keithweaver_/controlling-stepper-motors-using-python-with-a-raspberry-pi-b3fbd482f886 22/09/2018

Muhimu: <https://muhimu.es/medio-ambiente/botella-algas-biodegradable/>
2/07/2018

National Geographic: https://www.nationalgeographic.com.es/mundo-ng/actualidad/espana-planea-prohibir-plasticos-solo-uso_12598 24/07/2018

Programo Ergo Sum: <https://www.programoergosum.com/cursos-online/raspberry-pi/244-iniciacion-a-python-en-raspberry-pi/que-es-python> 17/09/2018

Raspberry Pi: <https://www.raspberrypi.org> 1/09/2018

Realfooding: <https://realfooding.com/bebidas-azucaradas/> 16/09/2018 i
1/11/2018

Reuters: <https://www.reuters.com/article/us-europe-environment-plastics/eu-moves-to-ban-single-use-plastics-idUSKCN1IT160> 25/07/18 27/11/2018

ScienceDirect:
<https://www.sciencedirect.com/science/article/pii/S0002870316300230>
(28/11/2018)