



C.A.R.L.E.S.

Card-Playing Algorithmic Robot for Leisure and Elder Socializing

PROJECT SPRINT #5.

DATE: 26 May 2020

VÍCTOR SUÁREZ 1493402

ORIOL MORENO 1496663

JAN MOROS 1492333

ADRIÀ CARRASQUILLA 1492104

Table of Contents

| | |
|---|-----------|
| Project description | 1 |
| Hardware Scheme | 2 |
| Software Architecture | 3 |
| Amazing contributions | 8 |
| Extra components and 3D pieces | 10 |
| Modules | 16 |
| Visió | 16 |
| Simulation | 20 |
| Inverse kinematics algorithm and visualizer | 26 |
| Foreseen risks and contingency plan | 30 |

C.A.R.L.E.S.

Card-Playing Algorithmic Robot for Leisure and Elder Socializing

Project description

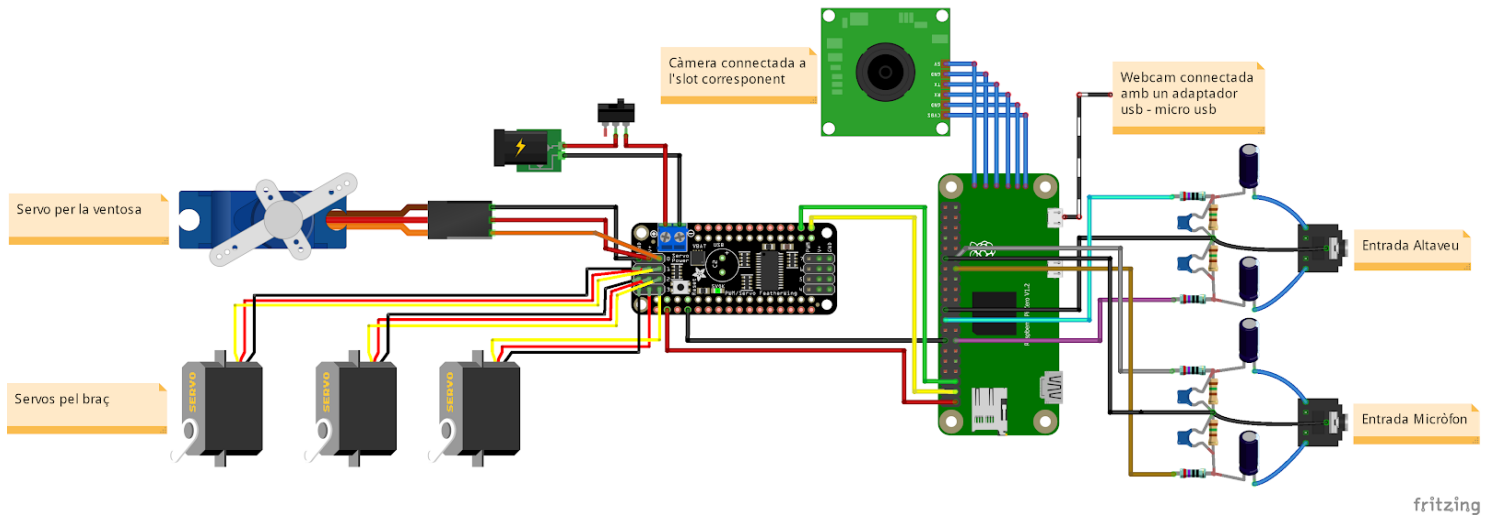
C.A.R.L.E.S és un robot capaç de jugar a partides lvs1 (Human vs AI) en el joc de la brisca.

El mecanisme principal consisteix en un braç antropomòrfic de 3 eixos, el qual permet un moviment en area circular sobre un tauler. A l'extrem de l'últim eix hi ha un manipulador format per una ventosa, la qual té controlada la seva pressió per un mecanisme pneumàtic amb un servomotor i una xeringa. El tauler s'adapta al recorregut del braç, tot i que mantenint al màxim la disposició típica del joc de la brisca. La mà del robot es troba a la seva dreta i de manera elevada per poder tenir una càmera a l'interior de la caixa que les sosté i així poder dur a terme un reconeixement de cartes per visió per computador. Les cartes de cada jugador, el trumfu i la baralla es troben al centre del tauler. També hi ha una estructura per tenir elevada una segona càmera que controla l'àrea de joc. Per últim, a l'esquerra del braç hi ha un mecanisme que permet al robot girar una carta.

Tot i això, aquesta part del robot no es podrà dur a terme per motius lògics en situació de pandèmia. Llavors el projecte se centra en desenvolupar exclusivament el software del robot i poder pulir cada mòdul de manera grupal i com a conjunt. En C.A.R.L.E.S és capaç de :

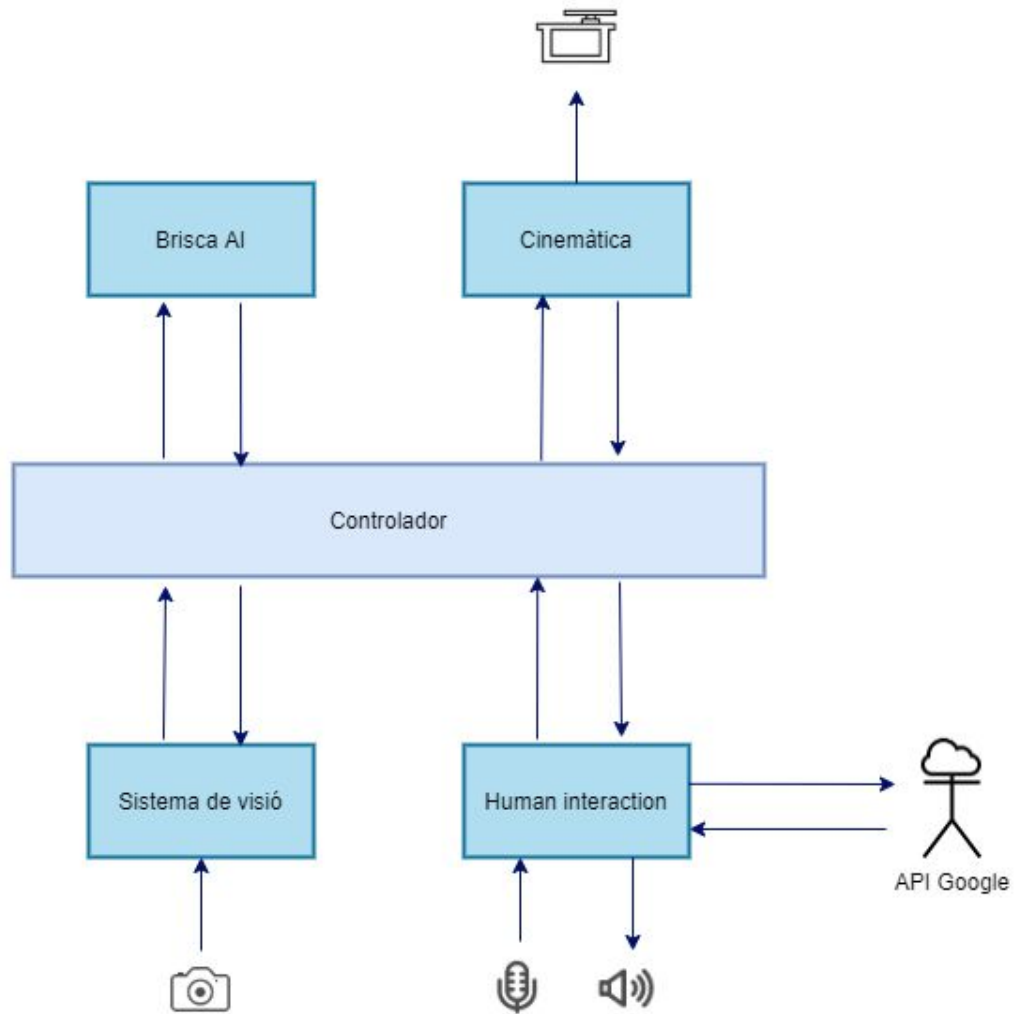
- Reconèixer cartes amb el mòdul de visió (número i pal).
- Dur a terme una partida de brisca amb un algorisme d'IA que intentarà guanyar al contrincant.
- Controlar les accions a realitzar (començar i acabar partida, robar carta, saber que toca canvi de torn, etc.) mitjançant comandes de veu que l'humà farà fer servir durant l'execució.
- Calcular els angles de rotació dels motors del braç per tal de moure el manipulador (inclòs el que controla la pressió de la ventosa) d'un punt a un altre, acord amb els moviments que hauria de realitzar en cas de la seva creació física.
- Unificació de totes les habilitats esmenades en un sol flux de treball, el qual hauria de ser el que tingués el robot físic.

Hardware Scheme



[Enllaç a la imatge completa](#)

Software Architecture



| | |
|-------------|---|
| Mòdul: | Brisca AI |
| Descripció: | <p>Model intel·ligent basat en heurístiques, que, tenint en compte:</p> <ul style="list-style-type: none"> - el pal al que va la partida - com d'avançada està aquesta - els punts que val cada carta - les cartes que ja s'han jugat - l'ordre en el que s'ha de jugar aquella ronda <p>Decidirà quina de les tres cartes que té a la mà ha de jugar, a partir d'un sistema d'heurístiques, basat en cost i benefici.</p> <p>El cost és el valor d'aquella carta multiplicat condicionalment per si és del mateix pal que la partida.</p> <p>El benefici, que només intervé a l'heurística si la IA és la segona en jugar aquella ronda, es calcula com els punts que aportaria a la IA guanyar la ronda, és a dir, la suma dels valors de les dues cartes. Si en canvi aquella ronda la guanyaria l'humà, aquesta suma és negativa.</p> <p>L'heurística final és el cost en el cas que jugui primer la IA o la resta ponderada del benefici menys el cost en el cas que ja hi hagi una carta sobre la taula.</p> <p>A més, l'algorisme anirà emmagatzemant les cartes a mesura que es juguin, i d'aquesta manera quan s'acabin totes les cartes de la pila, sabrà quines cartes té l'altre jugador, i en aquell moment es podria aplicar un algorisme minimax per decidir l'ordre en el que jugar les seves.</p> |
| Entrades: | L'estat de la partida en aquell instant: pal al que va, mà, nombre de jugades ja realitzades, llista de cartes jugades, puntuació actual dels jugadors, cartes sobre la taula, jugador que comença. |
| Sortides: | L'estat de la partida després del moviment escollit. |

| | |
|-------------|---|
| Mòdul: | Detecció de cartes mitjançant visió per computador |
| Descripció: | Algorisme que a partir d'una imatge detectarà les cartes i les identificarà per saber les seves cartes i les cartes jugades. Farem una cerca de contorns per detectar la carta, la transformarem a perspectiva frontal i reconeixem el nombre de la carta mitjançant template matching, per identificar el pal contarem els contorns dels marges. |
| Entrades: | Imatge |
| Sortides: | Una llista amb 3 cartes(Parella de valor i pal). La carta jugada per l'altre jugador. A l'inici de la partida llegim la primera carta que dirà el pal de la partida. |

| | |
|-------------|--|
| Mòdul: | Cinemàtica |
| Descripció: | <p>Aquest mòdul s'ha dividit en dos parts, una per les cinemàtiques del nostre robot exactes i l'altre per aprendre a fer inverse kinematics:</p> <p>1ª part:</p> <p>Esquelet de com funciona el C.A.R.L.E.S. si es connectes amb arduino i existís un projecte físic</p> <p>Algorismes i funcions que a partir de les mesures del robot i l'entorn ja conegudes, permeten manipular el robot per a realitzar les següents accions:</p> <ol style="list-style-type: none"> 1.Anar a una posició x-y-z 2.Agafar amb la ventosa 3.Deixar anar amb la ventosa 4.Girar la carta d'una posició definida a una posició definida <p>2ª part:</p> <p>Algoritme que permet visualitzar l'animació de qualsevol translació que vulgui fer el C.A.R.L.E.S.</p> <p>0.Es mostra el braç en la posició inicial per defecte</p> |

| | |
|-----------|--|
| | <p>1.Introducció d'unes coordenades</p> <p>1.1.Introduïdes per l'usuari</p> <p>1.2 Proposades per el programa</p> <p>2.L'algorisme divideix la translació en moviment molt petits ja que així controlem la velocitat i no fem moviments poc intuïtius per l'humà que estigui davant.</p> <p>Per cada moviment que completerà la translació l'algorisme fa:</p> <ol style="list-style-type: none"> Es realitzen els càlculs geomètrics per obtenir els angles en que s'ha de colocar el braç per arribar a tal posició Es calcula una visualització de com seria tal posició del braç segons els angles calculats (Per mostrar la visualització es fan uns càlculs per similitud de triangles i utilitzant el seno i el coseno dels angles obtinguts trobar les coordenades on comença i acaba cada part del braç) Es mostra la visualització de on i com esta el braç exactament. <p>Realitzem aquesta operació per cada moviment fins a completar la traslació.</p> <p>En quan aquesta ha finalitzat passem a una nova traslació o es finalitza el programa.</p> |
| | Posició x,y,z on es vol traslladar el braç |
| Sortides: | Animació de la traslació final |

| | |
|-------------|---|
| Mòdul: | Human Interaction |
| Descripció: | <p>Algorisme basat en l'API de voice recognition que ofereix Google. S'utilitza per analitzar les comandes citades per la veu del contrincant. Accions possibles:</p> <ul style="list-style-type: none"> - Començar partida <ul style="list-style-type: none"> - Preguntar nom contrincant - Preguntar edat - Finalitzar partida amb confirmació - Fi partida: avisar i tornar a començar o finalitzar. - Indicar al robot que és el seu torn - Indicar al robot que ja pot robar <p>Diagrama d'estats: idle: abans de començar jugant una carta de la mà</p> |
| Entrades: | Senyal d'àudio captada pel micròfon |
| Sortides: | Codi de l'acció a realitzar |

| | |
|-------------|---|
| Mòdul: | Controlador |
| Descripció: | <p>Mòdul principal que connectarà la resta de mòduls entre si per a unificar la funcionalitat del robot.</p> <p>Estats:</p> <ol style="list-style-type: none"> 1. Inicialització: reposicionament dels motors. Missatge de benvinguda. 2. Idle esperant comanda inici partida. Preguntar nom i edat. 3. Inici partida. Demanar a l'humà que reparteix-hi les cartes i espera fins a que tot estigui llest. 4. Missatge gràcies per indicar que el robot serà sempre el primer en tirar. 5. Observar el triomfu. 6. Comprovació: té tres cartes a la mà, la pila de cartes és al seu lloc,. 7. Inici partida. Bucle principal (mentre quedin cartes a la pila): <ol style="list-style-type: none"> 7.1. Mirar les nostres Cartes 7.2. El primer jugador tira una carta |

| | |
|-----------|--|
| | <p>7.3. El segon jugador tira una carta</p> <p>7.4. Si ha guanyat el robot, recull les cartes guanyades i les desa a la pila de guanyades. Si no, avisa a l'humà de que les reculli i espera una comanda que li digui que ja ha recollit les cartes guanyades.</p> <p>7.5. El jugador que ha guanyat la ronda roba</p> <p>7.6. L'altre jugador roba</p> <p>8. Es repeteix el mateix amb les últimes tres cartes, però sense robar</p> <p>9. Fi de la partida: Avisar a l'humà dels punts acumulats per cada jugador i qui ha guanyat</p> <p>10. Preguntar a l'humà si vol tornar a jugar. Si respon afirmativament, tornar al 3. Si no, tornar al 1.</p> |
| Entrades: | Cap |
| Sortides: | Cap |

Amazing contributions

Les contribucions increïbles que aporta el nostre projecte al Món son:

1. Entreteniment per a persones de la tercera edat
2. Automatització de Joc clàssic, la brisca
3. Voice recognition amb human interaction

A continuació explicarem amb més detall cada una d'elles:

1. Entreteniment per a persones de la tercera edat: El nostre projecte s'obre en el mercat de l'entreteniment dedicat a un tipus de client que el mercat tecnològic, ara per ara, no demanda. Mentre la majoria de humans estem cada cop més inmersos en una era digital, la gent de la tercera edat encara no té tants serveis tecnològics preparats per ells i per tant el nostre robot jugador de cartes, seria una alternativa novedosa per elles en quan a entreteniment. A més, és un joc que ja coneixen i l'únic factor que canvia es que no juguen contra un humà.

2. Automatització de Joc clàssic, la brisca : El nostre robot donarà vida a un dels grans jocs de sobretaula català i espanyol, la brisca. La tecnologia crea i innova, nosaltres li donem força a un joc del passat clàssic perquè perduri, mirem al futur recordant el passat. Una de les possibles millores del C.A.R.L.E.S. es que pugui jugar

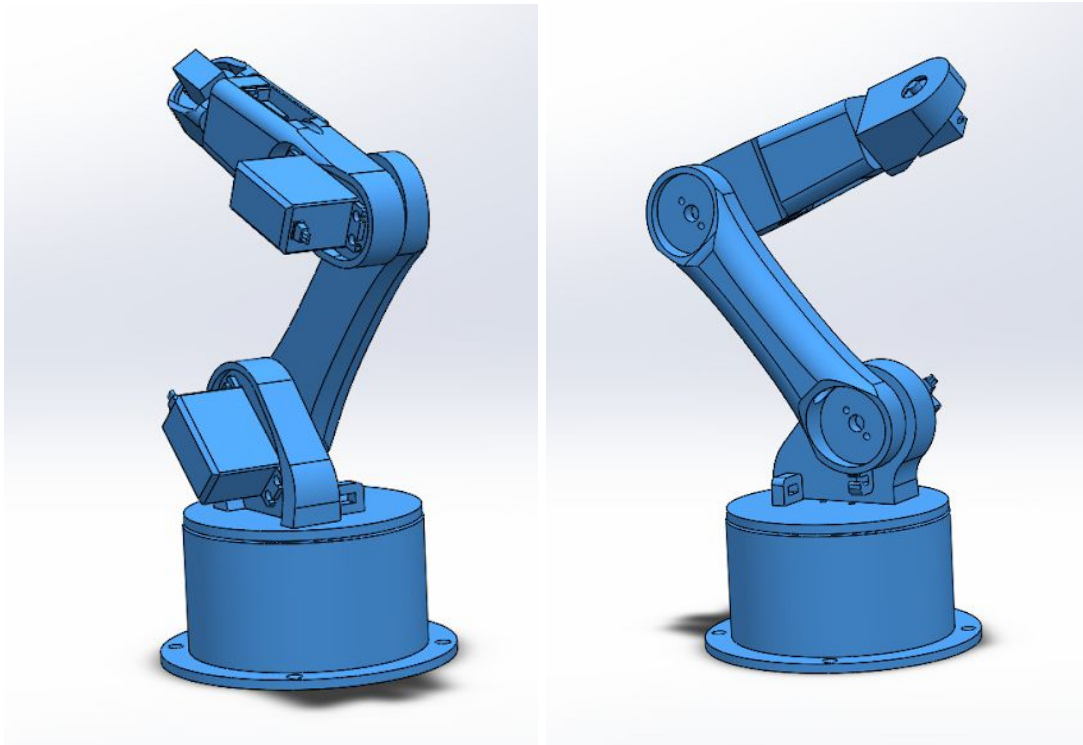
a més jocs clàssics de cartes a part de a la brisca, convertint-se en un jugador de tot tipus de jocs de cartes i tornant a posar de moda el jugar a cartes.

3. Voice recognition amb human interaction: El programa estarà preparat perquè, mitjançant un sistema de voice recognition, pugui parlar amb el contricant, una persona. Però no només parlarà per indicar moviments o exposar questions referents al joc, sino que intentarà connectar amb la persona. Des de preguntar coses per conèixer la persona (per guardar informació com el nom), fins a dir alguna broma que li alegri una mica el dia (per exemple, referir-se a una iaia com joveneta). Aquest mòdul l'implementarem primer en una versió més senzilla en la qual només ens comuniquem amb la persona per qüestions del joc però l'objectiu és millorar-lo si el temps ens ho permet fins al punt que a la persona li pugui resultar estimulante parlar amb en C.A.R.L.E.S.

Extra components and 3D pieces

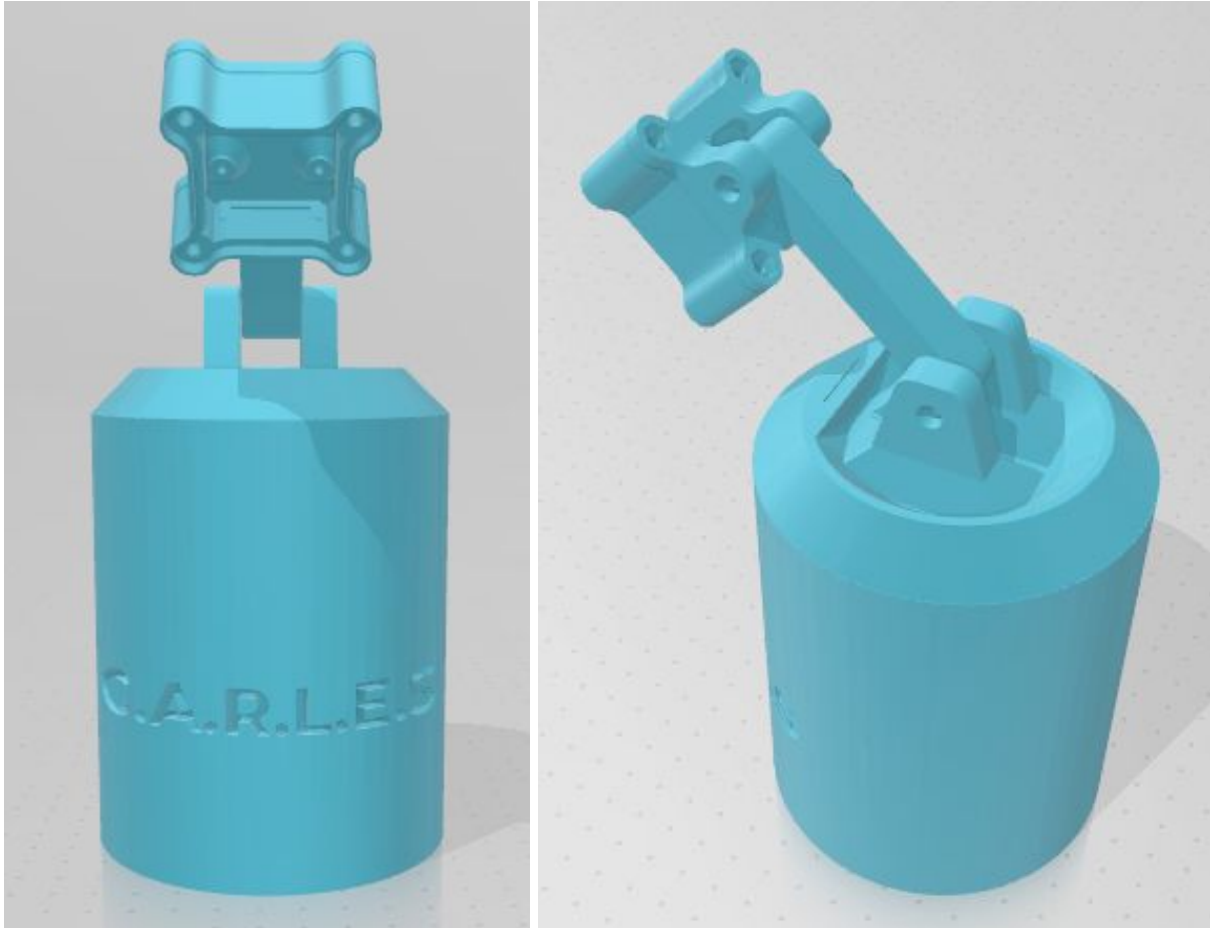
■ Braç del robot:

Braç antropomòrfic de 3 eixos.



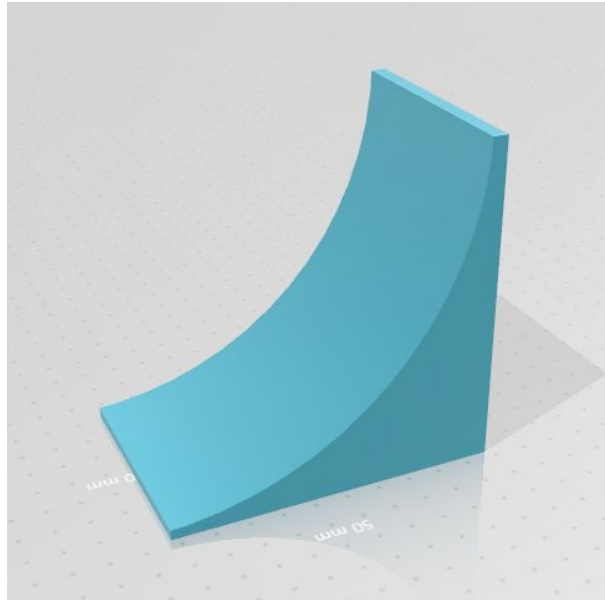
■ **Suport per la càmera del tauler:**

Suport en per subjectar la càmera que apunta al tauler de joc des d'una posició fixa.



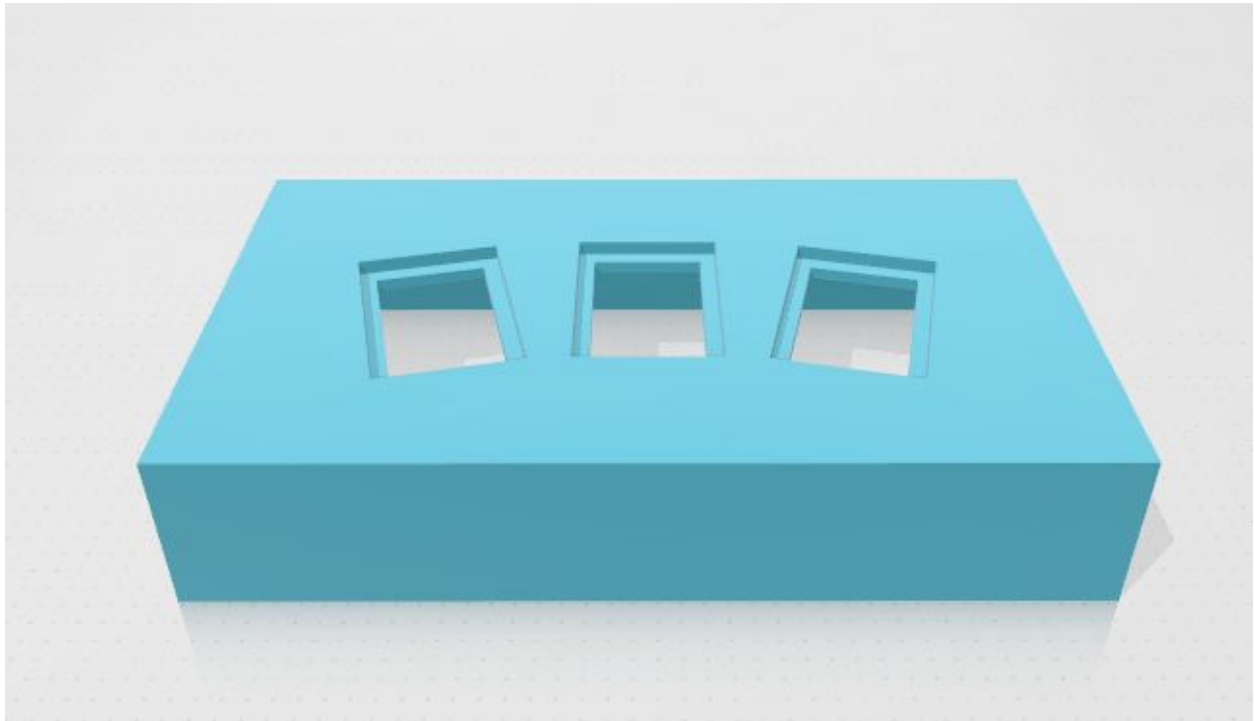
■ Girador de cartes:

Aquest es tracta d'una petita rampa a partir de la qual el braç és capaç de girar la carta.



■ Caixa que serà la mà de cartes del robot:

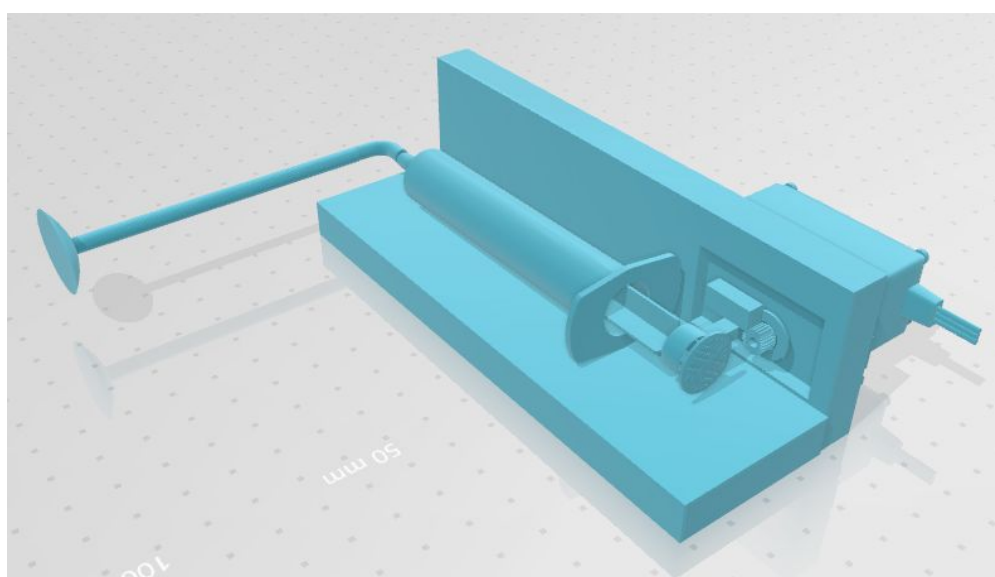
Caixa amb tres slots perquè el robot hi posi les cartes que formen la seva mà. Aquests slots estan buits de tal manera que soporten la carta i alhora permeten que una càmera que hi haurà situada a l'interior sigui capaç de detectar de quina carta es tracta.



■ Ventosa + xeringa:

Utilitzarem un mètode curiós per agafar les cartes, que vam escollir per temes econòmics. Es tracta d'una xeringa que connectada a un tub de plàstic que acaba amb una ventosa. Al tirar de l'èmbol de la xeringa xuclem l'aire de la ventosa creant-hi un buit que ens permetrà adherir-nos a qualsevol superfície. Així doncs activant l'èmbol (l'activarem amb un servo) activarem la ventosa per agafar qualsevol objecte. Aquesta idea l'hem extret d'un projecte que posem a continuació:

<https://www.youtube.com/watch?v=jQXTxGTjyes>

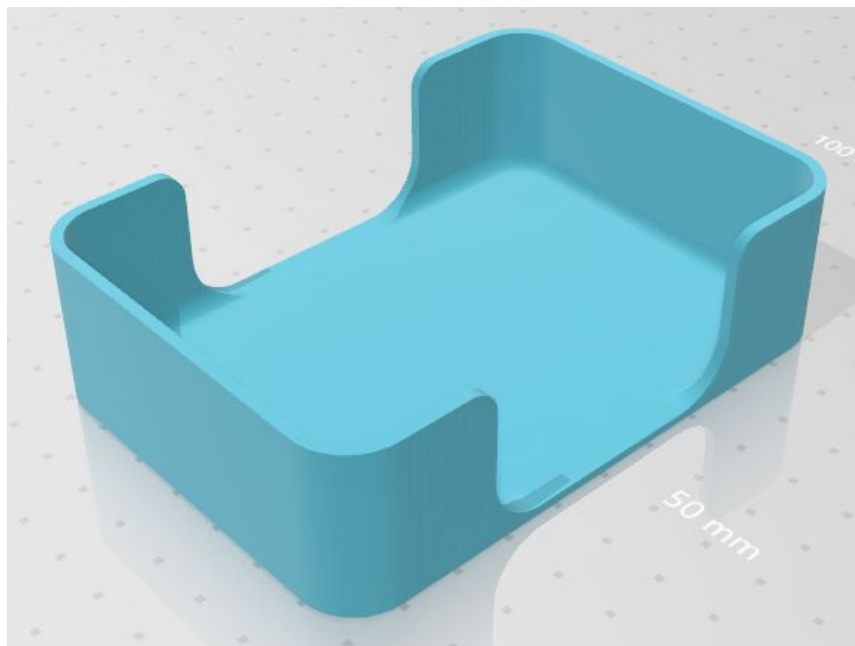


■ Tauler:

Tauler de joc on van subjectes la resta de peces, i que a més conté diversos slots per posar-hi el triomfu i les cartes tirades tant pel C.A.R.L.E.S. com per l'humà. Aquesta peça l'haguéssim fet manualment, de fusta.

■ **Caixetí per la baralla i per les cartes guanyades:**

Caixetí més gran que els slots del tauler, per poder-hi deixar més cartes. Això és necessari per la pila de robar, i per la pila de cartes guanyades pel C.A.R.L.E.S..

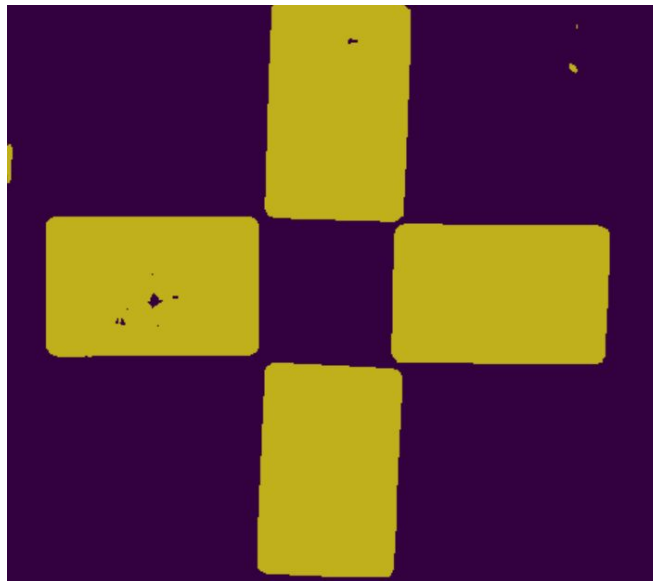


Modules

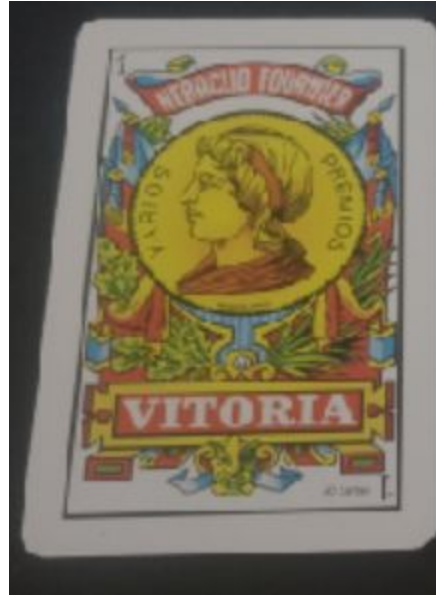
Visió

Com s'ha vist en l'arquitectura software, aquest mòdul s'encarrega de detectar les cartes i identificar-les.

Per al procés de detecció, en primer lloc s'aplica un threshold a la imatge per separar les cartes del fons.



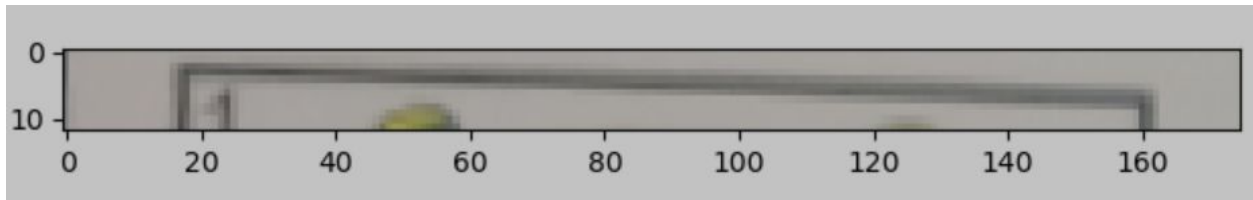
Aleshores s'utilitza un detector de contorns i es descarten aquells contorns que no corresponen a les característiques d'una carta (rectangulars, contorns "pares", ...) i es retalla de la imatge original l'interior d'aquests contorns.



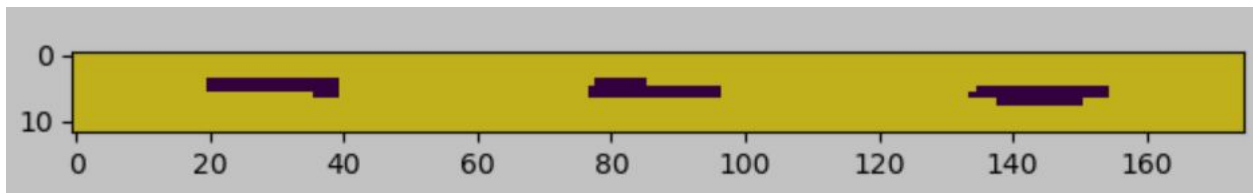
Com es pot apreciar, degut al punt de vista de la càmera les cartes es poden veure deformades, això dificultaria la identificació d'aquesta. Per a solucionar això, s'aplica una transformada de perspectiva per veure la carta tal i com si s'observès desde dalt. El resultat és el següent:



Una vegada s'ha detectat les cartes, es procedeix a identificar-les. Per a fer això es retallen les nostres regions d'interès tant per detectar el número com el pal. Pel que fa a la detecció del pal utilitzem la següent regió:



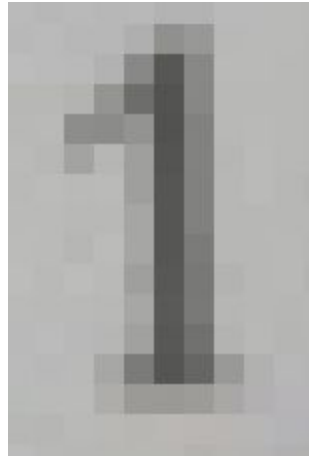
El marge de la carta ens aporta la informació del pal depenent del número de línies tenim un pal o un altre en aquest cas una sola línia representa una carta d'ors. Per tant es realitzaran transformacions morfològiques per aïllar aquest marge i detectar el pal comptant les línies del marge.



D'altra banda pel que fa a la detecció del numero s'utilitza una altre regió d'interès:



Aleshores es compara amb *template matching* aquesta regió amb unes imatges obtingudes d'avant mà que contenen els números de les cartes:



Finalment el *template matching* aporta un percentatge de similitud de cada nombre amb la carta, aquell amb un percentatge més alt és el número de la carta. Ara ja es disposa del número i el pal de cada carta, que serà utilitzat per el mòdul Brisca AI.

Simulation

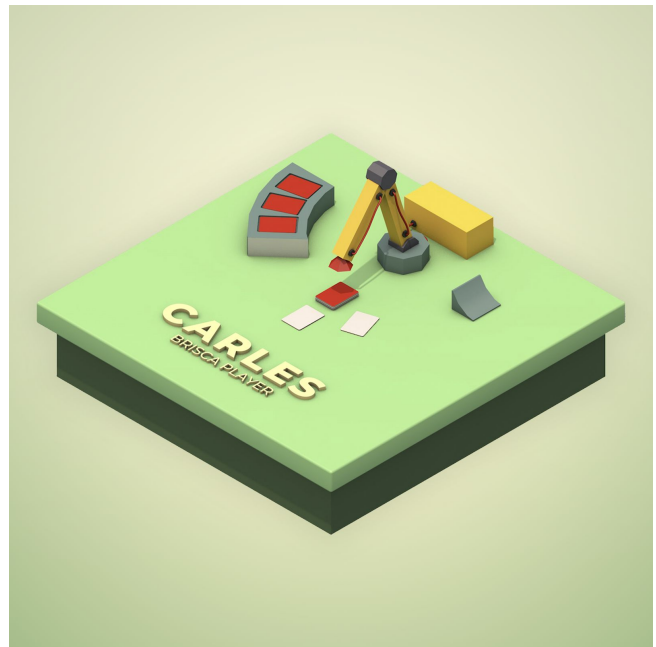
Donada la situació de pandèmia en la que es va dur a terme aquest projecte, no es va poder realitzar la construcció física del robot però es va optar per implementar un espai de simulació per recrear el comportament d'aquest.

La part més visible d'aquest projecte és la part mecànica que juntament amb el mòdul d'intel·ligència artificial (i evidentment, la resta), permeten dur a terme una partida de brisca. És per això que s'ha decidit fer una simulació principal en la IA. Els mòduls de visió i voice recognition són força independents per si mateixos i afegir-los a la simulació és un pes afegit innecessari. En canvi, el mòdul de cinemàtica sí que podria haver estat incorporat en la mateixa simulació. Tot i això s'ha optat de fer una simulació el màxim de rica pel que fa a aspecte visual. Les eines que ens permetien tal objectiu dificultaven cohesionar aquests dos conceptes sota el mateix projecte, per tant per una banda s'ha simulat el càlcul de l'inverse kinematics amb animacions de Python i per l'altra, s'ha simulat aquest moviment amb animacions dissenyades per nosaltres mateixos.

Per tant, aquesta simulació principal consta de:

- Un disseny 3D del model que es volia construir
- Una sèrie d'animacions que corresponen a les calculades amb inverse kinematics i que anaven a dirigir tots els moviments del robot real
- Un entorn Unity que ha convertit el que era una IA modular en un videojoc interactiu 3D.

La imatge que es mostra a la dreta és el model final que s'ha utilitzat com a escena dins de la simulació a Unity. Ha estat modelat i animat dins de Cinema 4D, un software específic per aquesta tasca. Gracies a aquest entorn hem pogut dotar d'una inverse kinematics al nostre model de manera automàtica i així obtenir un efecte visual molt més complet. Evidentment, el càlcul real no s'ha descartat doncs com ja s'ha comentat aquest mòdul té la seva propia demostració i simulació.



Un cop dins l'entorn Unity, el qual està pensat per a videojocs en dos i tres dimensions, s'importa aquest model per l'escena i s'implementa la intel·ligència artificial per a poder dur a terme una partida. Donat que aquest mòdul havia estat dissenyat en python i Unity funciona en C#, s'ha hagut de realitzar una migració per no haver de fer comunicacions externes a l'entorn.

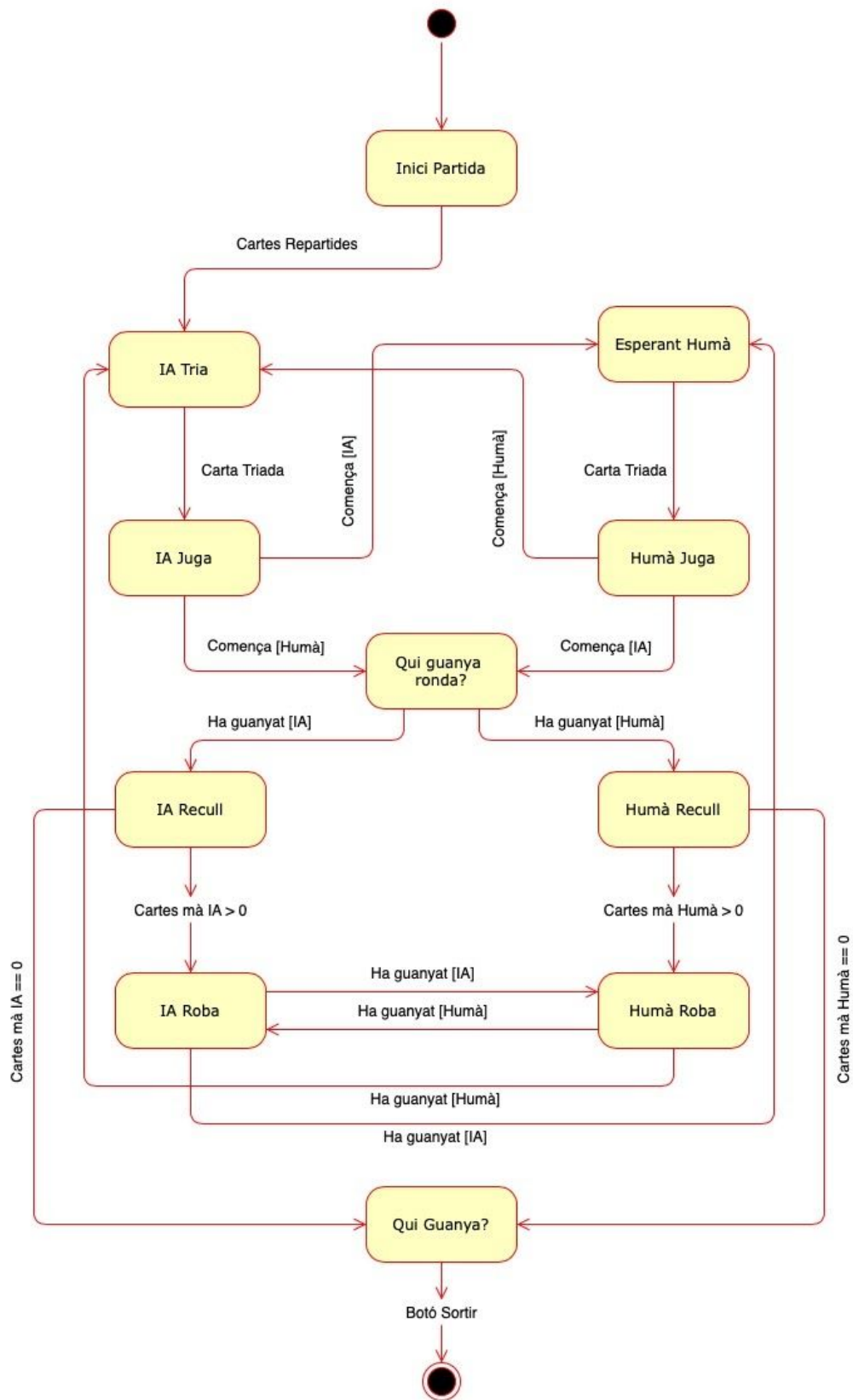
El joc, aleshores, et permet fer una partida lvs1 contra el robot CARLES (tal i com es pretenia fer en la versió física) de manera que el jugador tindrà visibles les seves cartes a jugar i el robot actuarà en funció al que l'usuari faci.

Quan parlem de videojocs, entenem que tenim un component principal que s'actualitza (si s'escau) cada frame. Per tant tot el codi es manté en un bucle constant. Per adaptar-nos a aquesta situació s'ha decidit dissenyar la simulació com una màquina d'estats, on cada estat és un instant de la partida on es poden realitzar diverses accions tal i com en la versió original.

Aquest estats són:

- Inici partida: S'agafa la primera carta de la baralla i es col·loca a la posició de la pinta, i les sis següents que passen a formar les mans d'ambdós jugadors. Per a totes aquestes accions hi ha una animació, que s'activa a mesura que es van fent.

- IA tria: Es crida a la funció que calcula, amb la informació que hi ha en aquell moment sobre la taula, quina de les cartes de la mà s'ha de triar. Ho fa a partir del sistema d'heurístiques explicat a la software architecture.
- IA juga: El braç mou (tant internament com a l'animació) la carta seleccionada de la mà a sobre la taula.
- Esperant a l'humà: Espera a que el jugador humà pitgi un dels botons, seleccionant així la carta de la mà que vol jugar.
- Humà juga: Es crida l'animació on l'humà juga una carta i es posiciona aquesta a sobre la taula.
- Qui guanya ronda: Fa els càlculs necessaris per determinar quin dels dos adversaris ha guanyat aquesta ronda. S'activa un clip d'àudio aleatòriament d'entre dues llistes, on en C.A.R.L.E.S. s'anima o s'enfada segons el vencedor de la ronda.
- IA recull: El braç recull de sobre la taula les cartes que acaba de guanyar, i les sosa al capdamunt de la pila de cartes guanyades. Es sumen els valors d'aquestes a la seva puntuació.
- Humà recull: Es crida l'animació on l'humà recull de sobre la taula les cartes que acaba de guanyar. Es sumen els valors d'aquestes a la seva puntuació.
- IA roba: El braç agafa la primera carta de la pila i la posa a l'slot que tingui lliure de la seva mà.
- Humà roba: Es crida l'animació on l'humà agafa la primera carta de la pila, i s'actualitza el text corresponent al HUD perquè l'usuari sàpiga quina carta acaba de robar.
- Qui guanya joc: Estat final de la màquina, es calcula quin dels dos jugadors ha obtingut major puntuació, es crida l'animació de victòria o derrota del braç i el clip d'àudio corresponent.



Tal i com s'ha explicat, gairebé tots els estats comparteixen un parell de detalls comuns: Criden una animació del robot i actualitzen les dades que permeten mantenir el fluxe del joc. A més es permet tenir un control de l'input de l'humà.

Per tant amb aquesta simulació es cobreix directament el mòdul d'intel·ligència artificial, l'inverse kinematics és l'equivalent a la crida d'animacions i l'input del jugador amb botons juntament amb el control de les dades generades artificialment es considerarien l'equivalent de la funció que realitzarien els mòduls de visió i voice recognition. Dit això podem reafirmar que cobreix totes les funcionalitats que preteníem implementar amb el robot físic.

A continuació es mostren diferents moments possibles al jugar una partida en aquest entorn.



Aquesta és la pantalla que es mostra al iniciar el joc. Permet pitjar el botó jugar per iniciar la partida. La resta d'elements es mostraran durant la partida.



Durant la partida es mostra el número de cartes que queden a la pila, la pinta de la partida i les cartes sobre la taula, si n'hi ha com a dades de joc. A més el jugador tindrà 3 botons, un per cada carta, amb el contingut d'aquesta. Quan és el seu torn només ha de clicar-la per posar-la en joc. A més, amb la simple finalitat de mostrar el procés hi ha darrera executant-se, a la cantonada superior dreta es mostra informació general de la partida.



En aquest cas el jugador ha perdut 98 a 22 contra en C.A.R.L.E.S. Té la opció de sortir del joc. En cas de victòria es mostra una pantalla similar amb el missatge corresponent.

Cal destacar que, seguint el concepte que vam definir com amazing contribution de crear un robot amb una utilitat especial per a persones d'edat avançada que volen gaudir d'una possible companyia en un joc de cartes, la temàtica del joc ha estat ambientada en aquest concepte també. S'ha fet un processat d'imatge de manera que la pantalla sembla la d'un televisor antic (distorsió de lent, granulat i aberració cromàtica). A més per a fer-ho més agradable s'ha incorporat una melodia d'ambient que acompanya l'escena de 1vs1.

Inverse kinematics algorithm and visualizer

Aquest modul ha sorgit després del covid per entrar al àmbit del càlcul del inverse kinematics, te aquest objectius:

1. Calcular els angle a partir d'unes posicions x,y,z finals del braç exactes
2. Mostrar una simulació 3D de com es veuria el braç
3. Mostrar una animació de com realitzar el moviment de translació d'unes posicions x,y,z a unes posicions finals x,y,z

La primera part concisteix en el calcul de Invers Kinematics purs.

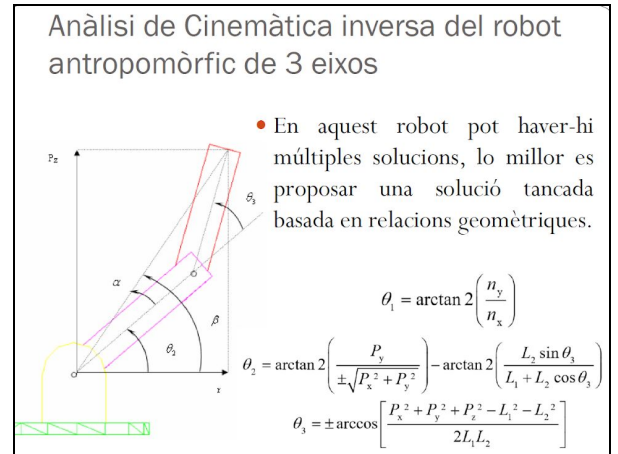
1.Calcul de inverse kinematics:

Em calculat els angles a través de mètodes geomètrics. Aquests càlculs els em fet de dues maneres en el transcurs a la finalització del projecte:

Primerament, com inicialment no vam trobar informació per internet o a l'assignatura de com fer aquests càlculs exactes per el nostre tipus de robot, les vam calcular nosaltres.

Paper i llapís, fòrmules de sinus, cosinus i de similitud de triangles i a calcular. Ha estat un procés interessant per el fet de tornar a fer càlculs matemàtics com a la ESO o al Batxillerat.

Però teníem algun petit error que no ens calculaba tots els possibles cassos. Per sort, vam preguntar al professorat de l'assignatura i ens van preparar una sessió per nosaltres on ens donava les fòrmules geomètriques exactes per el nostre projecte i doncs ens em apollat en aquestes.



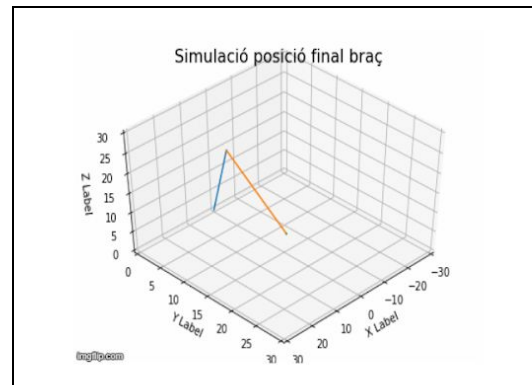
Afegir que la sessió que ens va passar el Carlos l'hem penjat al guithub perquè creiem vital per a dur a terme la cinemàtica del nostre C.A.R.L.E.S.

2.Simulació del braç

Per a dur a terme aquesta simulació volíem agafar algun programa d'Internet que ens ho calcules però finalment vam decidir crear-ho nosaltres amb els coneixements que ja tenim de programació. Per això hem fet servir la llibreria matplotlib.

Ha sigut molt complicat aconseguir aplicar els angles ja calculats a aquest entorn creat des de zero, però finalment hem obtingut un resultat acceptable.

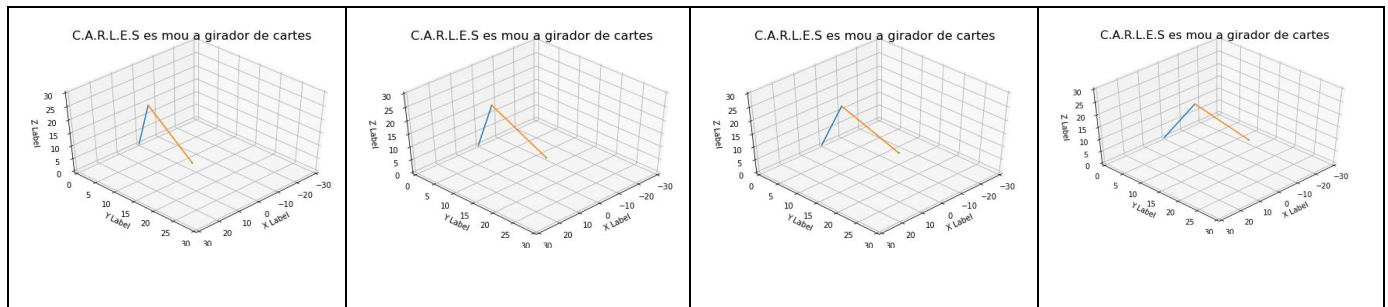
Visualment no té moltes llums ja que la part més visualment bonica l'hem destinat a la simulació 3D, aquest mòdul volia ser funcional.



3.Animació de la translació del braç d'anar unes posicions x,y,z a unes x,y,z finals

Es una animació feta amb el mateix programa que calcula els angles a partir de unes posicions x,y,z però a més a més mostra la translació que faria en C.A.R.L.E.S. per anar d'una posició a un altre.

Els resultats els podem veure al guithub o a la presentació final o a la carpeta de la entrega realitzada. Adjuntem 4 frames perquè es vegi evolució:



Per a dur a terme aquesta part hem tingut hem estudiat les següents qüestions que ara comentarem breument:

- Limitacions dels servos: Els servos tenen unas limitacions, i cal tenir en compte quines son les seves característiques:
 - Pes màxim que poden transportar,
 - la repetibilitat
 - la carga nominal
 - la velocitat màxima dels servos
 - Limits d'angles màxims que pot realitzar cada servo

Com el nostre robot només agafa cartes d'una en una que tenen un pes quasi nul, i a més a més no es convé que els servos vagin massa ràpid ja que està pensat per jugar contra persones de la tercera edat que pot ésser que no estiguin acostumats a les noves tecnologies i doncs un robot que fa moviments molt ràpids, bruscs i poc intuïtius no ens convé.

Així doncs les limitacions dels servos no ens suposen ni suposarien cap problema greu, a priori.

- Anàlisis d'esforç o estrès + control d'errors: Aquí no hem pogut entrar per falta de model físic, però seria un apt important del projecte en quan a temps si la part física la tiéssim endavant.
- Obtenir més d'una solució per un conjunt d'angles: Aquest possible problema hem descartat fent els càlculs geomètricament sense contemplar possibles opcions múltiples.

- Jacobiana: No ens ha calgut calcular la matriu jacobiana perquè no volíem velocitat ni teníem càlculs per matrius
- Tipus de trajectòria: Respecte al tipus de trajectòria hem utilitzat una molt intuïtiva per a l'hora, com es la *trajectòria rectilínia contínua*.
Per aconseguir aquest camí hem de trobar les coordenades (x, y, z) entre les coordenades d'inici i de referència final que estan en la mateixa línia, aquests punts es calculen amb la mateixa interpolació lineal amb la qual s'extreuen punts finits d'un camí rectilini de referència. Les articulacions per tant es mouen de forma isòcrons. I per això emplearíem la llibreria VarSpeedServo .h, la qual hem vist a problemes però no em tocat degut a que es part de la part física.

Foreseen risks and contingency plan

Descartats els riscos de la part hardware, per el canvi de pla en el projecte pel coronavirus; ja no tenim problemes considerable. Ara tenim aquests possibles problemes de software pero que si seguim treballant com ara tots ells es resolen.

| Risk # | Description | Probability (High / Medium / Low) | Impact (High/ Medium / Low) | Contingency plan |
|--------|--|--|--------------------------------------|---|
| 1 | Detecció de la carta malament | Low | Medium | ---- No cal contenció ja que tenim el mòdul ben avançat. En tot cas, en el pitjor dels casos ens faria perdre la partida, cosa que no és important ja que l'objectiu principal no és guanyar la partida sino jugar-la (human interaction) |
| 2 | Derrota constant de la IA | Low | Low | ---- L'objectiu principal no és guanyar la partida sino jugar-la (human interaction), però hi ha pensades certes millores a l'algorisme, si resulta no complir amb les expectatives |
| 3 | Mòdul d'àudio tan senzill que la human interaction sigui insatisfactoria | low | low | ---- El mòdul està construït de manera que s'hi poden afegir més instruccions, i més maneres diferents de que les digui l'humà. |

References

This project has been inspired by the following Internet projects:

Projecte principal en el que ens em inspirat:

https://www.societyofrobots.com/robot_arm_card_dealer.shtml

Altres projectes:

<https://www.youtube.com/watch?v=jQXTxGTjyes>

<https://www.youtube.com/watch?v=wNPzzvskyzU>

<https://www.youtube.com/watch?v=9e8gvJnXpjY>

<https://docplayer.es/25310256-Inteligencia-artificial-en-el-juego-de-la-brisca.html>