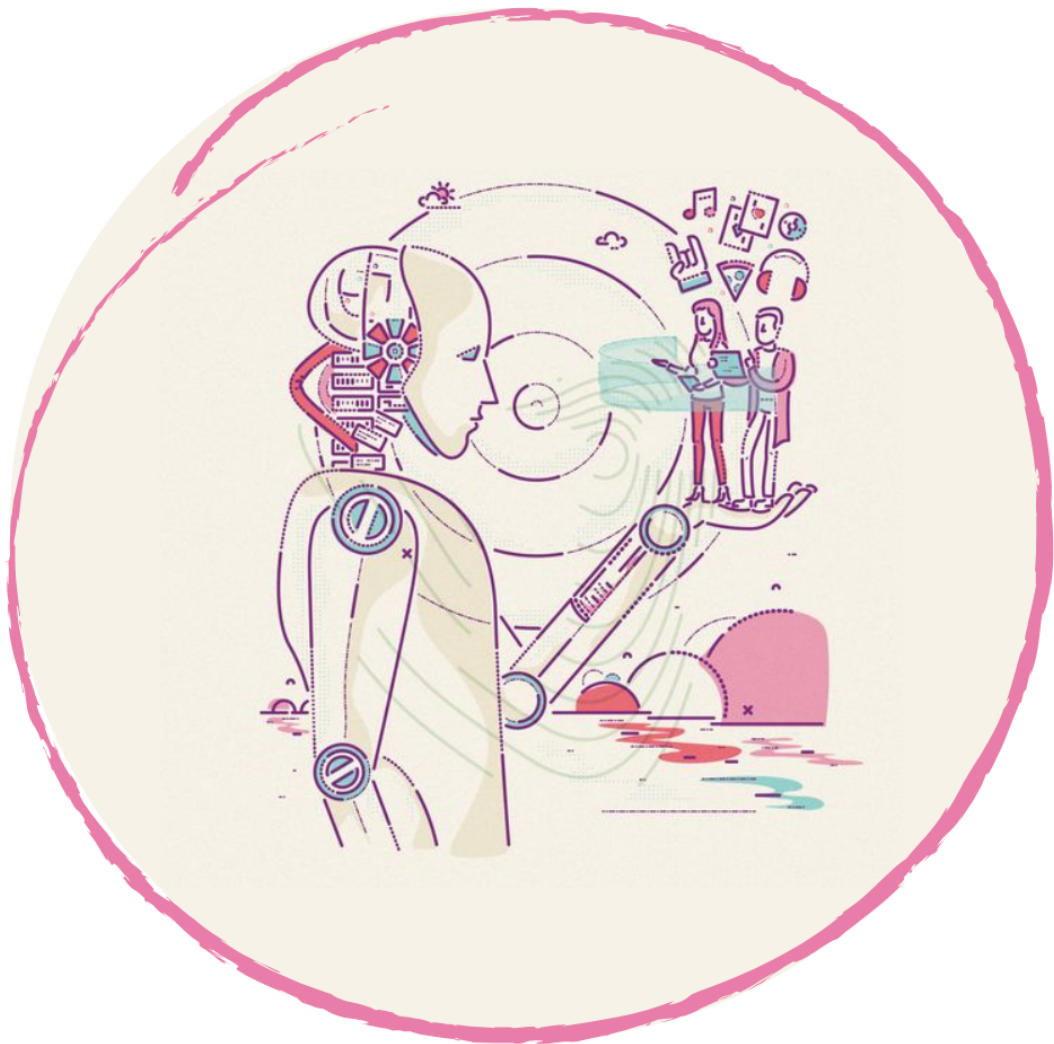


LA INTEL·LIGÈNCIA ARTIFICIAL AL MUSEU NACIONAL ARQUEOLÒGIC DE TARRAGONA



Pseudònim: VaquitaFeliz

ÍNDEX

	PÀG
1. INTRODUCCIÓ.....	3
1.1. MOTIVACIÓ.....	3
1.2. OBJECTIUS I HIPÒTESI.....	4
1.3. METODOLOGIA.....	5
2. OBJECTIUS DE DESENVOLUPAMENT SOSTENIBLE.....	6
3. MARC TEÒRIC.....	7
3.1. INTEL·LIGÈNCIA ARTIFICIAL.....	7
3.1.1. APRENENTATGE AUTOMÀTIC.....	9
3.1.2. APRENENTATGE PROFUND.....	10
3.2. PROCESSAMENT DE LLENGUATGE NATURAL.....	11
3.2.1. TOKENITZACIÓ.....	11
3.2.2. LEMATITZACIÓ.....	12
3.2.3. ELIMINACIÓ DE PARAULES BUIDES.....	12
3.3. XATBOT.....	13
3.3.1. QUÈ ÉS UN XATBOT?.....	13
3.3.2. TIPUS DE XATBOTS.....	14
3.3.3. ARQUITECTURA D'UN XATBOT.....	16
3.4. RECERCA SOBRE LA LLOBA CAPITOLINA.....	18
4. MARC PRÀCTIC.....	22
4.1. TECNOLOGIES.....	22
4.1.1. LLENGUATGE DE PROGRAMACIÓ PYTHON.....	22
4.1.2. ENTORN DE DESENVOLUPAMENT IDLE.....	23
4.2. APLICACIÓ D'ESCRITORI.....	25
4.2.1. ÚS DE L'APLICACIÓ D'ESCRITORI AL MUSEU.....	25
4.2.2. FLUX DE NAVEGACIÓ.....	27

4.2.3. CREACIÓ DE PANTALLES.....	29
4.3. XATBOT.....	32
4.3.1. FITXER DE DADES.....	32
4.3.2. PREPROCESSAMENT DE DADES.....	33
4.3.3. CREACIÓ DE DADES D'ENTRENAMENT.....	34
4.3.4. CONSTRUCCIÓ DEL MODEL.....	34
4.3.5. PREDICCIÓ DE RESPOSTES.....	35
4.3.6. FINESTRA DEL XATBOT.....	36
4.3.7. INTEGRACIÓ DEL XATBOT A L'APLICACIÓ D'ESCRITORI.....	36
4.3.8. EXECUCIÓ DEL XATBOT.....	37
5. CONCLUSIONS.....	39
5.1. ASSOLIMENT D'OBJECTIUS.....	40
5.2. PROBLEMES I SOLUCIONS.....	42
5.3. PUNTS DE MILLORA.....	44
6. BIBLIOGRAFIA I WEBGRAFIA.....	46
7. ANNEXOS.....	48
7.1. FITXER DE DADES DEL XATBOT.....	48
7.2. CODI PER A ENTRENAR I CREAR LA INTEL·LIGÈNCIA ARTIFICIAL DEL XATBOT.....	49
7.3. CODI DE L'APLICACIÓ D'ESCRITORI I DE LA FINESTRA DEL XATBOT.....	50

I. INTRODUCCIÓ

I.I. MOTIVACIÓ

Abans de començar el meu Treball de Recerca (TdR) m'agradaria explicar el que em va motivar a escollir el tema de la intel·ligència artificial.

Primer de tot, sempre m'ha agradat tot allò que té a veure amb la tecnologia i els dispositius electrònics. A mesura que m'he anat fent gran el meu interès en aquesta àrea ha anat creixent. A 1r de Batxillerat vaig haver d'escollir entre diferents optatives, però una d'elles em va cridar més l'atenció que les altres. Aquesta era l'optativa de programació. Quan vam començar em va semblar interessant com funcionava la programació, encara que vam començar amb el més bàsic. En acabar el curs vaig obtenir una sèrie de coneixements per programar un joc, una calculadora i fins i tot una aplicació. D'aquesta manera, el meu interès per la tecnologia va augmentar.

Quan vam haver de fer la tria de temes per al TdR, tenia clar que volia fer la meva recerca basada en la tecnologia. La meva professora de programació em va proposar fer un treball orientat a les intel·ligències artificials (IA) i això em va semblar una molt bona idea. Em va motivar perquè penso que el tema de treballar una intel·ligència artificial és molt interessant i innovador per a tots. En l'actualitat, la intel·ligència artificial està cada vegada més present i l'objectiu és fer-nos la vida més fàcil.

Llavors, com que la intel·ligència artificial és molt àmplia, és a dir, abasta molts àmbits, vaig haver de reflexionar sobre com la podria utilitzar. Així doncs, vaig pensar a fer una intel·ligència experta en la meua ciutat natal, Tarragona. Però encara faltava acotar més el tema del treball. Fent una recerca sobre Tarragona, vaig veure que una IA experta en una ciutat era un tema molt extens de fer, com: la cultura, els costums, la gastronomia, festes majors i festes tradicionals, llocs per visitar, museus, restes romanes, parcs d'atraccions, parcs aquàtics, parcs naturals, platges, rutes de muntanyes, etc.

Donada la importància que té Tarragona com a ciutat Patrimoni de la Humanitat per les seves restes arqueològiques romanes, declarada el 30 de novembre de

l'any 2000, vaig decidir centrar-me en el Museu Nacional Arqueològic de Tarragona (MNAT). Vaig investigar sobre aquest museu i vaig pensar que podria fer una aplicació d'escriptori per als visitants que incorporés un xatbot que els ajudés a obtenir respostes a les preguntes que es fessin sobre les obres i escultures del museu. D'aquesta manera vaig escollir, finalment, que el meu TdR aniria sobre construir una aplicació d'escriptori basada en el MNAT amb un assistent virtual expert (xatbot) en la Lloba Capitolina, que és una de les escultures més importants del museu i també una de les figures clau en la mitologia romana.

Per tant, la idea del meu TdR és que els visitants, a l'entrada del museu, agafin una tauleta tàtil on hi hagi l'aplicació d'escriptori instal·lada, i a mesura que es vagin movent pel museu, puguin preguntar-li dubtes al xatbot. L'aplicació és un prototip, el xatbot només respon preguntes sobre la Lloba Capitolina, i no sobre altres obres i escultures del museu, però es podria ampliar perquè ho fes.

I.2. OBJECTIUS I HIPÒTESI

OBJECTIUS

Per fer el meu treball m'he marcat quatre objectius per assolir. Són els que indico a continuació:



El primer objectiu és investigar què és una intel·ligència artificial i poder crear-ne una sense grans coneixements, i demostrar que es pot utilitzar el machine learning o aprenentatge automàtic per fer un assistent virtual expert en qualsevol tema.



Per tal de construir una intel·ligència artificial m'he proposat aprendre sobre els xatbots per programar-ne un i per adaptar-lo a les necessitats del MNAT.



D'altra banda, per fer ús d'aquesta intel·ligència artificial, dissenyaré i construiré una aplicació d'escriptori per poder ajudar als visitants del

MNAT a tenir una visita guiada pel museu amb un xatbot que pugui resoldre els seus dubtes.



L'últim objectiu del meu treball és buscar informació sobre una obra d'art del MNAT, en aquest cas la Lloba Capitolina, perquè el xatbot pugui respondre els dubtes dels visitants sobre aquesta obra a través d'un xat dintre de l'aplicació.

HIPÒTESI

Després d'haver escollit el tema de crear un xatbot per al MNAT, vaig investigar sobre museus importants.

Fent la recerca vaig veure que efectivament hi ha museus on ja s'ha implementat aquesta guia virtual. Per exemple, al Museu Nacional d'Art Africà i al Museu Hirshhorn (EUA) s'usen guies virtuals amb l'objectiu de resoldre els dubtes dels visitants.

Per tant, això em va generar una pregunta: es podria introduir un assistent virtual al MNAT per ajudar els visitants del museu en el seu recorregut? La resposta d'aquesta pregunta és sí, i ho demostraré en aquest treball de recerca, construint un xatbot expert en una obra d'art: la Lloba Capitolina.

I.3. METODOLOGIA

En aquest apartat explicaré la metodologia i els passos que vaig seguir per fer la part teòrica i la part pràctica del meu TdR.

Per començar, vaig fer una investigació prèvia sobre el MNAT i sobre la Lloba Capitolina. Després vaig pensar totes les possibles preguntes i respostes que poden fer-se els visitants.

Amb tota aquesta informació ja podia construir el xatbot. Llavors, primer de tot vaig fer el cervell del bot on estan totes les preguntes i respostes. Per aconseguir que funcionés correctament vaig entrenar la IA fent-li preguntes

perquè donés les respostes adequades als visitants, i així cada cop fos més intel·ligent. Per tant, vaig fer el codi de la finestra on està el xatbot. Aquesta serà la part on els usuaris podran interactuar amb el guia virtual.

Per altra banda, vaig construir l'aplicació d'escriptori que s'utilitzarà a les tauletes del museu. En obrir l'aplicació hi ha una pàgina de benvinguda. Després apareixen algunes de les obres més importants del museu. Allà, hi ha un botó per a cada obra d'art on he integrat la finestra del xatbot perquè s'obri. Com aquest treball és un prototip, només he fet el xatbot de la Lloba Capitolina. L'última pàgina inclou una enquesta perquè les persones que facin servir l'aplicació puguin donar la seva opinió i possibles millores sobre la seva experiència al museu.

2. OBJECTIUS DE DESENVOLUPAMENT SOSTENIBLE

El meu TdR treballa els següents Objectius de Desenvolupament Sostenible (ODS) de l'agenda 2030:



Imatge 1: ODS que es treballen al TdR



El primer objectiu és l'ODS 9, que parla sobre innovació. Aquest objectiu pretén construir infraestructures, promoure la industrialització sostenible i fomentar la innovació. Per tant, amb la meva aplicació estic promovent la innovació mitjançant una guia virtual a un museu.



El segon objectiu de l'agenda que es treballa és l'ODS 10, reducció de les desigualtats. La meva aplicació promou aquest objectiu, ja que permet ajudar a persones amb discapacitats físiques permetent-los gaudir del museu com qualsevol altra persona. Per exemple, una persona amb discapacitats auditives no pot fer una visita guiada,

però amb el xatbot pot resoldre els seus dubtes escrivint i llegint. També una persona que té dificultats de mobilitat pot descobrir les obres d'art del museu amb l'aplicació.



I finalment, l'objectiu número 4. Aquest ODS parla sobre una educació inclusiva i de qualitat per promoure oportunitats d'aprenentatge per a tothom. Podem dir que el meu TdR treballa aquest objectiu perquè garanteix que tots els visitants del museu puguin arribar a la informació de qualsevol obra d'art.

3. MARC TEÒRIC

3.1. INTEL·LIGÈNCIA ARTIFICIAL

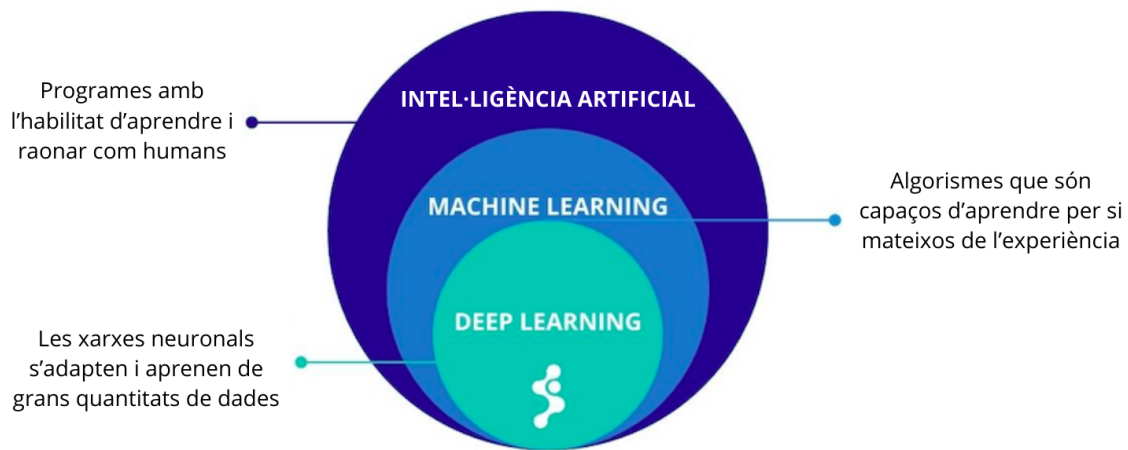
Què és la intel·ligència artificial? Doncs la intel·ligència artificial o IA és un concepte pel qual encara no existeix una definició acceptada globalment. Segons Tim Cook, CEO de l'empresa Apple, “La intel·ligència artificial és la tecnologia més transformadora de la nostra era. Té el potencial de canviar la forma en la qual vivim, treballem i ens relacionem amb el món”.



Imatge 2: Usos quotidians i futurs de la IA

Però també podem dir que la IA és l'habilitat de les màquines per realitzar les capacitats dels humans, com el raonament, l'aprenentatge, la creativitat i la capacitat de plantejar. És un camp de la informàtica que s'enfoca a crear sistemes que poden fer feines que normalment requereixen la intel·ligència humana.

La IA permet a la tecnologia percebre el seu entorn, relacionar-se amb ell, resoldre problemes i actuar amb un fi específic. La intel·ligència artificial és capaç d'adaptar-se a comportaments per analitzar els efectes d'accions prèvies i treballar de manera autònoma. La IA està dissenyada pels éssers humans per a un objectiu determinat i actua en la dimensió física o digital.



Imatge 3: Tipus d'intel·ligència artificial

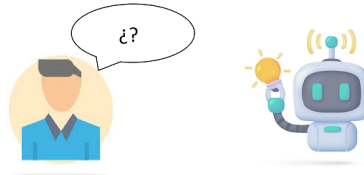
3.1.1. APRENTATGE AUTOMÀTIC

L'aprenentatge automàtic o machine learning (ML) és una branca de la intel·ligència artificial que permet a les màquines aprendre dels errors per millorar el seu funcionament automàticament amb els intents. En lloc de fer servir una sèrie de normes rígides (programació de regles), els models d'aprenentatge automàtic s'adapten amb l'experiència i els exemples.

Hi ha diferents tipus d'aprenentatge automàtic. Un d'ells és l'aprenentatge automàtic supervisat que utilitza models que s'entrenen amb blocs de dades i etiquetes. El model s'entrena utilitzant dades de les quals ja es coneix la sortida

correcta. La part de supervisió la fa un humà, que a cada dada li assigna una etiqueta.

En el cas del xatbot, l'humà defineix totes les preguntes possibles que pot fer l'usuari i les respostes associades, però el seguiment de la precisió i l'ajustament de l'algoritme es realitza de manera automàtica pel mateix algoritme.

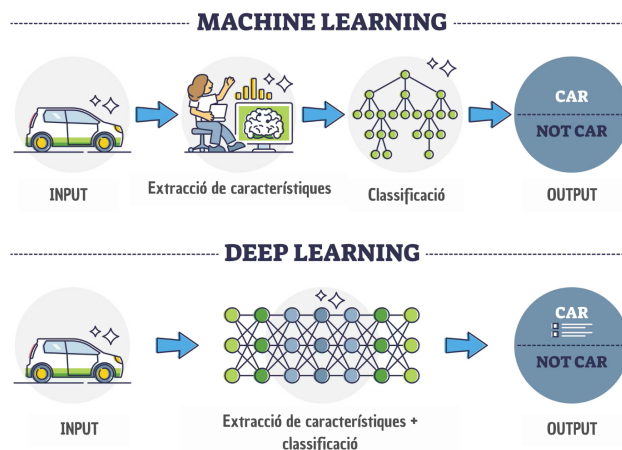


Imatge 4: Usuari interactuant amb la IA

Per exemple, si tenim un conjunt de preguntes i respostes i l'usuari introdueix una pregunta diferent de les que estan dintre del cervell del xatbot, el model d'aprenentatge automàtic utilitzarà totes les dades anteriors per predir l'etiqueta de la pregunta. Això vol dir que si introduïm una pregunta nova sense etiqueta, el model endevinarà si la pregunta és d'un bloc o d'un altre, predint així la resposta que ha de donar el xatbot.

3.1.2. APRENTATGE PROFUND

L'aprenentatge profund o deep learning (DL) està dintre del grup d'aprenentatge automàtic, però utilitza xarxes neuronals.



Imatge 5: Diferència entre com funciona l'aprenentatge automàtic i el profund

3.2. PROCESSAMENT DE LLENGUATGE NATURAL

El processament de llenguatge natural (PLN) és la branca de la IA que s'ocupa de les interaccions entre màquines i el llenguatge humà. Aquest ajuda les intel·ligències artificials a analitzar, processar i generar el llenguatge humà. Se centra en el processament de les comunicacions humanes, dividint-les en parts per identificar els elements més rellevants del missatge. Els guies virtuals o xatbots són els més coneguts per utilitzar-ho.

Però aquest processament de llenguatge natural no fa més intel·ligent al xatbot, sinó que li dona la capacitat de processar i generar llenguatge humà.

El processament de llenguatge natural té diferents funcions que s'expliquen a continuació.

3.2.1. TOKENITZACIÓ

La tokenització permet a la IA descompondre les oracions o frases en paraules individuals, símbols i signes de puntuació, fent així unitats més petites i manejables, anomenades tokens. Aquesta funció fa un procés d'anàlisi semàntic que selecciona la paraula amb més sentit o importància del context donat. Per exemple:

- L'usuari de l'aplicació (visitant) introdueix una pregunta:
“De quin material està feta la Lloba Capitolina?”
- El xatbot processa l'oració i la descompon:
“De”, “quin”, “material”, “està”, “fet”, “la”, “Lloba”, “Capitolina”, “?”
- En aquesta frase, la paraula que té més importància és: “material”

3.2.2. LEMATITZACIÓ

La funció de lematització permet al xatbot reduir les paraules a la seva forma d'arrel o significat, eliminant així la seva forma gramatical. Per exemple:

Paraules: cantant, cantava, cantàvem, cantaven.

Totes aquestes paraules tenen la mateixa arrel "cantar". La lematització les agrupa en la mateixa forma canònica "cantar".

En conclusió, aquesta tècnica és essencial perquè ajuda a comprendre i processar el llenguatge d'una forma més precisa i eficient.

3.2.3. ELIMINACIÓ DE PARAULES BUIDES

Aquesta funció elimina les paraules comunes com articles, preposicions i pronoms que no aporten cap significat a l'anàlisi. Aquestes paraules són conegudes en el llenguatge natural, però per a la intel·ligència artificial no tenen valor informatiu.

Exemples de paraules buides:

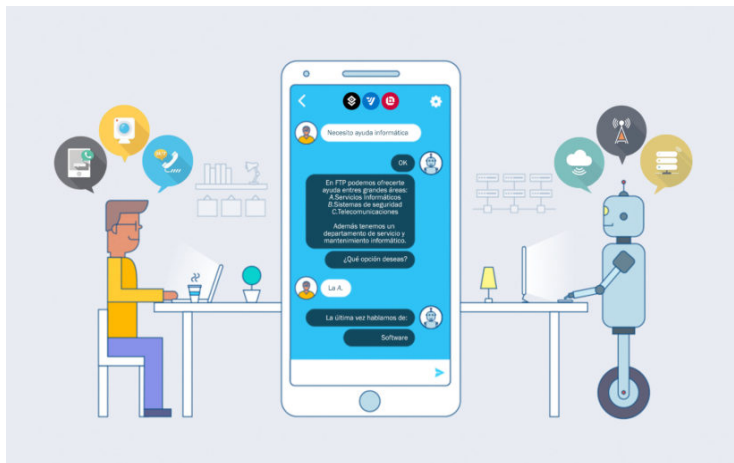
- "i"
- "el"
- "de"
- "la", "els", "les"
- "un", "una"
- "que"
- "en"

En conclusió, l'eliminació de paraules buides és una tècnica simple, però molt efectiva, ja que millora l'eficiència i la precisió del PLN.

3.3. XATBOT

3.3.1. QUÈ ÉS UN XATBOT?

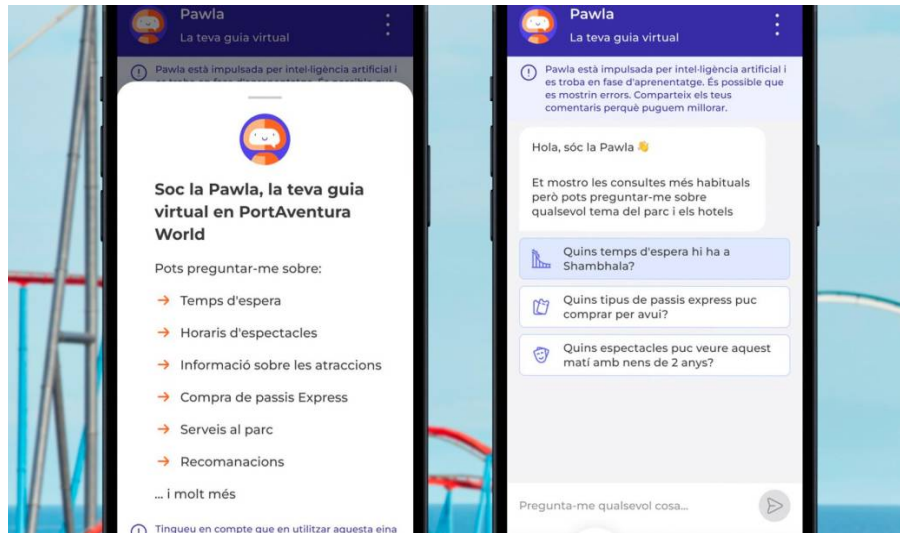
Un xatbot o bot és un programa informàtic o aplicació de software que simula la conversa humana amb un usuari i una IA mitjançant xats de text. Els xatbots funcionen amb regles i intel·ligència artificial que permeten comprendre el llenguatge humà, conversar amb els usuaris i respondre amb informació adequada. No tots els xatbots utilitzen la IA, però cada vegada s'està implementant més juntament amb el processament de llenguatge natural per respondre coherentment les preguntes dels usuaris. Les empreses fan servir els xatbots per oferir serveis com l'atenció al client i proporcionar informació sobre els seus productes o serveis. Es troben habitualment en aplicacions de missatgeria, assistents virtuals i aplicacions de servei al client.



Imatge 6: Humà interactuant amb un xatbot

L'objectiu principal d'un xatbot és proporcionar als usuaris assistència o informació de manera intuïtiva i conversacional.

Per exemple, un xatbot integrat a una aplicació d'escriptori d'un parc d'atraccions pot respondre preguntes freqüents i guiar el visitant pel parc a través d'instruccions específiques.

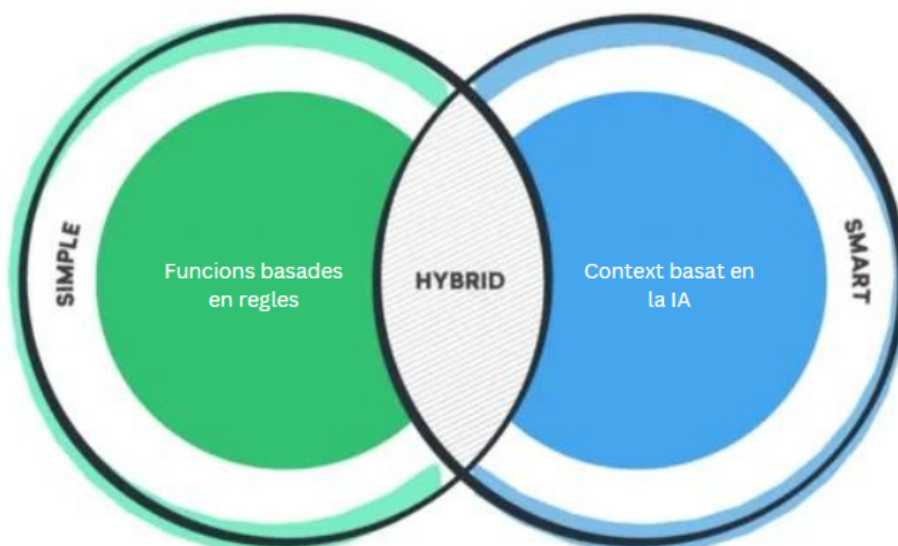


Imatge 7: Assistent virtual a PortAventura World

3.3.2. TIPUS DE XATBOTS

Els xatbots es poden dividir en 3 grups:

- Xatbots simples
- Xatbots híbrids
- Xatbots intel·ligents



Imatge 8: Tipus de xatbots

Xatbots simples

Els xatbots simples són aquells que tenen funcions limitades i estan basats en regles. Tenen uns deures predeterminats, el bot planteja preguntes basades en opcions limitades i el client pot escollir entre les opcions fins que obtingui la resposta adequada, però el xatbot no aprèn dels seus errors. Quan el bot interactua amb l'usuari, el flux del diàleg és fix. Aquests xatbots són adequats per converses simples i són molt fàcils de construir i entrenar.

Per exemple: demanar una hamburguesa.

El xatbot l'atendrà com ho faria un treballador per telèfon, realitzant preguntes sobre els ingredients, el tipus de carn, el punt de cocció, entre altres detalls. A continuació, li sol·licitarà la seva adreça i número de telèfon mòbil. Tot això es farà mitjançant opcions interactives que el client podrà seleccionar per completar la seva sol·licitud.

Xatbots híbrids

Els xatbots híbrids o hybrids són una combinació de xatbots simples i intel·ligents, situant-se en un punt intermedi entre ambdós. Aquests bots tenen algunes funcions basades en regles, però també poden comprendre la intenció del text de l'usuari. Això els converteix en una eina perfecta per empreses que volen interactuar amb els seus clients.

Per exemple: una consulta mèdica.

El xatbot li preguntarà a l'usuari els seus símptomes i en funció de les respostes, a través d'unes regles, li donarà un diagnòstic.

Xatbots intel·ligents

Els xatbots intel·ligents o smart utilitzen IA. Estan dissenyats per simular interaccions entre humans. Poden mantenir diàlegs fluids, dinàmics i poden comprendre la interacció, el llenguatge i el sentiment. A més a més, necessiten

programació per interpretar el context de les converses i resulten més complexos d'implantar i executar, ja que requereixen grans quantitats de dades per aprendre.

Per exemple: un assistent virtual en una pàgina web o l'Alexa o la Siri que comprenen el llenguatge humà mitjançant la veu.

3.3.3. ARQUITECTURA D'UN XATBOT

Un xatbot es divideix en diferents components per poder comprendre les preguntes dels usuaris, determinar les respostes adequades i interactuar convencionalment amb els humans. Els components essencials de l'arquitectura d'un xatbot són:

- La interfície d'usuari
- El motor de processament de llenguatge natural (PLN)
- L'administrador de diàleg
- La base de coneixement

Interfície d'usuari

Aquesta és la part visual on l'usuari pot interactuar amb la intel·ligència artificial a través d'una aplicació. La interfície d'usuari captura l'entrada de l'usuari, les seves preguntes, i mostra les respostes del bot.

Per aquest motiu, en l'elaboració d'aquest treball, vaig adonar-me que era necessari crear una finestra on l'usuari pogués interactuar amb el xatbot i integrar-la dins de l'aplicació.

Motor de processament de llenguatge natural

Aquest component és clau per a la IA, ja que permet la comprensió de converses amb el llenguatge humà.

Les tècniques del processament de llenguatge natural (PLN) permeten al xatbot:

- Analitzar l'estructura de l'oració per comprendre el llenguatge humà (anàlisi sintàctica).
- Identificar intencions (el que pregunta l'usuari) i entitats com per exemple la Lloba Capitolina (classificació d'intencions i extracció d'entitats).
- Determinar el context i el significat de la conversa.

Per aquest motiu, el meu xatbot havia d'incloure el processament de llenguatge natural per utilitzar aquestes tècniques.

Administrador de diàleg

L'administrador de diàleg determina la fluïdesa de la conversa i les respostes del xatbot en funció de les intencions i entitats extretes.

Aquest conté:

- Regles i fluxos de diàleg específics del domini.
- Frases desencadenants que activen el diàleg.
- Mapatge d'intencions a respostes a partir d'un model entrenat prèviament. S'assignen respostes a preguntes.

Per tant, vaig haver de crear dades d'entrenament i preparar a la IA perquè sigui capaç d'assignar respostes correctes a les preguntes formulades.

Base de coneixement

La base de coneixement conté dades específiques del domini que li donen al xatbot l'habilitat de tenir converses amb significat:

- Preguntes freqüents
- Manuals d'usuari
- Informació objectiva

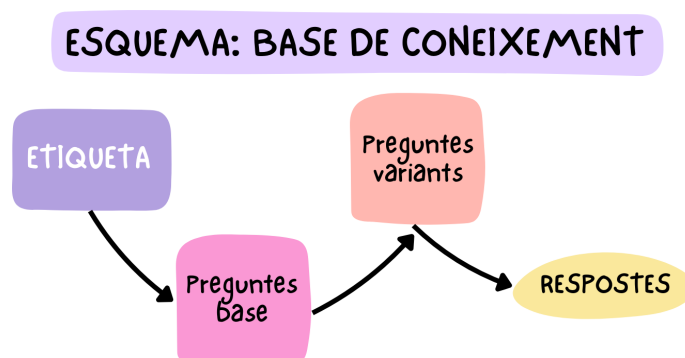
Aquestes dades estan codificades en formats com documents, en bases de dades, un fitxer JSON, etc. La base de coneixement conté essencialment el “cervell” del xatbot. Allà estan totes les preguntes i respostes del xatbot i es pot modificar per afegir-ne més.

En el meu treball, he utilitzat un fitxer JSON com a base de coneixement.

3.4. RECERCA SOBRE LA LLOBA CAPITOLINA

Per començar a fer la base de coneixement del xatbot, vaig posar-me en la pell d'un visitant del museu. Per aquest motiu, em vaig plantejar les preguntes que es podrien fer els visitants. Atès que el meu treball és un prototip, vaig centrar-me en la figura de la Lloba Capitolina, com he comentat a l'apartat 1.1, i vaig plantejar-me preguntes relacionades amb aquesta obra d'art.

Després de pensar totes les preguntes possibles, vaig buscar informació a internet per proporcionar al xatbot les respostes adequades.



Imatge 9: Esquema sobre la base de coneixement

Tenint totes les preguntes amb la informació corresponent, vaig seguir l'esquema de la imatge anterior per fer la base de dades.

Primer de tot, vaig pensar una pregunta base, i a partir d'aquesta vaig formular les possibles variants de preguntes que podrien fer-se els usuaris.

Per poder mostrar les preguntes i respostes del meu xatbot he fet una taula de classificació, que és la següent:

TAULA DE CLASSIFICACIÓ			
Etiquetes	Preguntes bases	Preguntes variants	Exemple resposta
SALUTACIONS	"Hola"	"Com estàs", "Hi ha algú?", "Bon dia", "Bona tarda"	"Hola, gràcies per parlar amb mi. Et puc ajudar en alguna cosa?", "Hola, visitant! Quins dubtes tens?"
FUNCIONS	"Quina és la teva funció?"	"Que fas?", "Que pots fer?", "En què em pots ajudar?", "Quin és el teu treball?", "Com em pots ajudar?"	"El meu objectiu és poder respondre a les teves preguntes sobre la Luperca o també anomenada Lloba Capitolina"
COMIAT	"Adéu"	"Fins després", "Fins a una altra", "Un plaer xerrar amb tu, adeu", "Fins a l'altra"	"Ha sigut un plaer parlar amb tu. Ja ens veurem!", "Espero poder parlar amb tu un altre cop. Que tinguis un bon dia"
PRESENTACIÓ	"Com et dius?", "Quin és el teu nom?"	"Em pots dir el teu nom?", "Tens algun nom?"	"Em dic BeepBoop i avui seré la teva guia virtual al Museu Nacional Arqueològic de Tarragona. Et puc resoldre totes les preguntes que vulguis sobre la Lloba Capitolina"
NO RESPOSTA	" "	" "	"Ho sento, no t'entenc", "Si us plau, dona'm més informació", "No estic segur d'entendre't"
MATERIALS I TÈCNICA	"De quin material està feta la figura?"	"Quina tècnica s'utilitza a la figura?", "Com és la tècnica que van utilitzar per fer-la?"	"La Lloba capitolina està feta amb bronze fos i utilitzant la tècnica de la cera perduda"
LLEGENDA DE L'ESTÀTUA	"Quina és la llegenda de la figura?", "Que diu la llegenda?"	"Com és la llegenda de la figura?", "Que diu la llegenda?", "Que representa l'estàtua?"	"Segons la mitologia, una lloba va trobar dos bessons, Ròmul i Rem, a la vora del riu Tíber. La criatura, contràriament al seu instint carnívor, va alletar i criar els dos nens que acabarien fundant la ciutat de Roma al turó del Palatino."

Imatge 10: Taula de classificació 1

Etiquetes	Preguntes bases	Preguntes variants	Respostes
ANYS DE CREACIÓ DE L'ESTÀTUA	"Quan es va fer la Lloba Capitolina?"	"En quin any es va crear l'estàtua?", "Quin va ser l'any de creació?"	"La Lloba Capitolina es va crear entre 1508 i 1513. Tradicionalment, es creia que la lloba va ser esculpida al segle V aC, durant l'època etrusca de Roma, mentre que les figures de Ròmul i Rem van ser afegides durant el Renaixement, probablement al segle XV."
AUTOR DE L'ESTÀTUA	"Qui va ser l'autor de la Lloba capitolina?"	"Qui va crear la Luperca?", "Com es diu l'autor de la lloba capitolina?"	"L'autor de la lloba capitolina és Antonio Pollaiuolo"
QUE REPRESENTA LA LUPERCA	"Que representa la Luperca?"	"Quina representació té la Lloba capitolina?", "Que significa la lloba?"	"La força femenina. La lloba és un dels animals espirituals més interessants i n'és una prova el simbolisme i el significat en diferents cultures. És un animal molt especial amb una energia molt potent de la que en moments de la nostra ens en podem beneficiar manifestant-la"
QUÈ ÉS?	"Què és la luperca?"	"Què és?", "La Lloba Capitolina que és?"	"La Lloba Capitolina és una estàtua de bronze que representa una lloba alletant dos nens, que segons la mitologia romana són Ròmul i Rem, els fundadors llegendaris de Roma."
IMPORTÀNCIA	"Per què és important la Lloba Capitolina per a la ciutat de Roma?"	"Quina importància té?", "Perquè és important"	"La Lloba Capitolina és un símbol de la fundació de Roma i del poder i la protecció de la ciutat. S'ha convertit en una icona cultural i representa l'origen mític de Roma."

Imatge 11: Taula de classificació 2

Etiquetes	Preguntes bases	Preguntes variants	Respostes
QUE REPRESENTEN ELS NENS	"Què representen Ròmul i Rem a l'escultura?"	"Quina representació tenen els nens?"; "Que signifiquen els nens a l'obra"	"Ròmul i Rem, els dos bessons lactants a l'estàtua, són personatges clau en la llegenda de Roma. Segons la tradició, Ròmul seria qui finalment fundaria la ciutat i es convertiria en el seu primer rei després de matar el seu germà Rem en una disputa."
L'ESTIL	"Quin és l'estil artístic de l'estàtua de la Lloba Capitolina?"	"Quin estil s'utilitza?"; "Que estil es fa servir?"	"L'estil de la lloba reflecteix la influència etrusca, amb una aparença ferotge i geomètrica. Les figures de Ròmul i Rem, en canvi, són d'un estil renaixentista més realista i detallat."
POLÍTICA I CULTURA	"L'estàtua ha estat utilitzada amb finalitats polítiques o culturals?"	"Quins fins culturals o polítics té?"; "S'ha utilitzat a la política o a la cultura?"	"Sí, la Lloba Capitolina ha estat utilitzada al llarg dels segles com a símbol de poder i unitat de Roma. Durant el règim feixista de Mussolini al segle XX, es va fer servir amb finalitats propagandístiques per connectar la Roma moderna amb la seva grandesa imperial passada."
MIDA	"Quina mida té la Lloba Capitolina?"	"Quant mesura l'estàtua"; "Quines mides té l'obra d'art?"	"La Lloba Capitolina té una alçada d'aproximadament 75 cm i una llargada de 114 cm. Les figures de Ròmul i Rem són molt més petites en comparació amb la lloba."

Imatge 12: Taula de classificació 3

4. MARC PRÀCTIC

4.I. TECNOLOGIES

4.I.I. LLENGUATGE DE PROGRAMACIÓ PYTHON

Python és un llenguatge de programació potent, fàcil d'aprendre, d'escriure i de llegir i a més es pot executar des de moltes plataformes diferents. S'utilitza en aplicacions web, en el desenvolupament de software i l'aprenentatge automàtic. També és perfecte per programar servidors, crear des de scripts senzills fins aplicacions complexes com intel·ligència artificial i anàlisi de dades.

Per fer una intel·ligència artificial experta en la Lloba Capitolina i l'aplicació d'escriptori per al museu MNAT, vaig analitzar diferents llenguatges de programació, i d'entre tots vaig escollir Python pels següents motius:

- Té una gran biblioteca que conté codi reutilitzable. D'aquesta manera els programadors no han d'escriure el codi des de zero.
- Hi ha molts recursos útils a internet per aprendre sobre aquest llenguatge. Per exemple, hi ha una gran varietat de vídeos tutorials, documentació i guies per a tothom.
- Tinc bons coneixements de Python que vaig adquirir en l'assignatura de programació de 1r de batxillerat.



Imatge 13: Logo de Python

4.1.2. ENTORN DE DESENVOLUPAMENT IDLE

IDLE (Integrated Development and Learning Environment) és l'entorn de desenvolupament oficial de Python amb codi obert i gratuït per totes les edats i nivells. Va ser creat per Guido van Rossum, el creador de Python per proporcionar una programació fàcil a l'hora d'utilitzar i executar programes en Python.

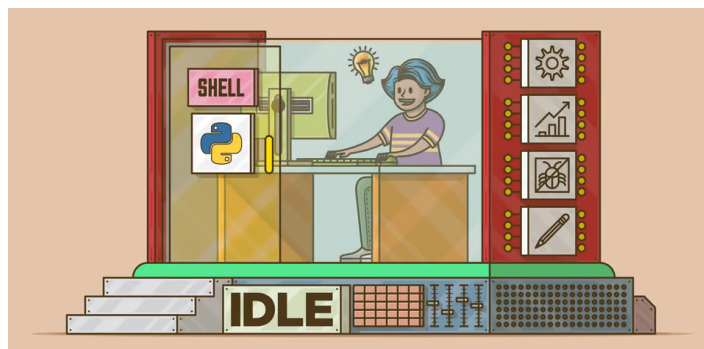
IDLE ofereix característiques avançades com la sintaxi, suggeriments de codi i autocompletat per facilitar l'escriptura de codi.

En obrir IDLE es pot començar a escriure el codi en l'editor de text. Llavors, com la consola de Python està integrada a la interfície d'IDLE, es pot executar el codi directament en la mateixa finestra.

Com a l'assignatura de programació de 1r de batxillerat ja havia treballat amb aquest entorn de desenvolupament, vaig decidir utilitzar-lo pel meu treball. Altres motius són:

- És molt fàcil d'usar i d'aprendre amb una interfície d'usuari senzilla.
- Proporciona un editor de text integrat amb funcions avançades d'edició de codi, que ajuden a detectar possibles errades.
- És compatible amb totes les versions de Python.

Per crear l'aplicació d'escriptori i el xatbot vaig haver d'importar un seguit de llibreries per tal que el producte final fos el més atractiu possible i funcionés correctament.



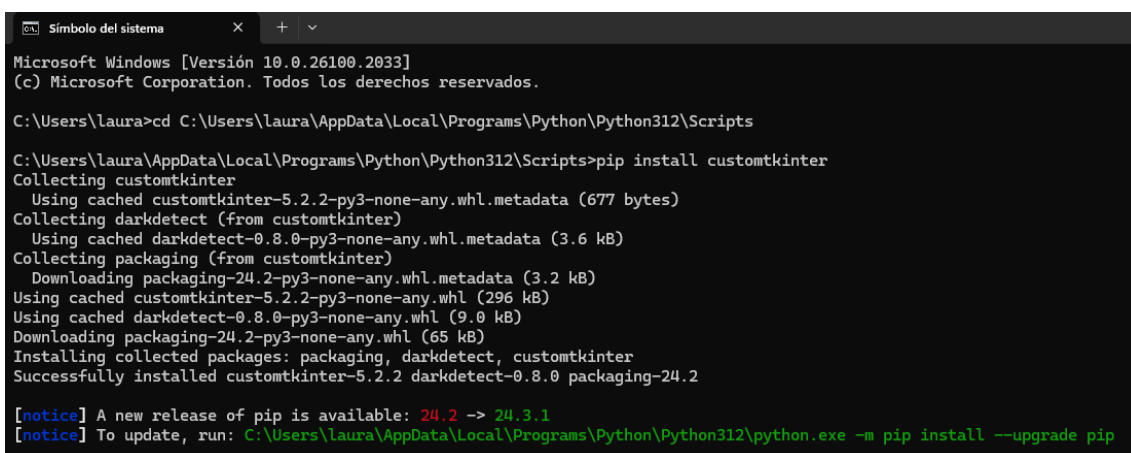
Imatge 14: Representació d'IDLE

LLIBRERIES

Perquè el codi de l'aplicació d'escriptori i el xatbot funcionin, vaig haver d'instal·lar aquestes llibreries de Python. Entre parèntesis indico l'àmbit de cada llibreria.

- CUSTOMTKINTER (interfície gràfica d'usuari)
- PILLOW (imatges)
- NUMPY (estadística)
- NLTK (PLN)
- TENSORFLOW (IA)
- KERAS (IA)

Primer vaig buscar la ruta on està l'aplicació PIP de Python. Després vaig obrir la consola del sistema de Windows (CMD) per instal·lar les llibreries i vaig canviar el directori posant la ruta que he dit abans. Finalment, vaig executar la instrucció “pip install + el nom de la llibreria”.



```
Símbolo del sistema
Microsoft Windows [Versión 10.0.26100.2033]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\laura>cd C:\Users\laura\AppData\Local\Programs\Python\Python312\Scripts

C:\Users\laura\AppData\Local\Programs\Python\Python312\Scripts>pip install customtkinter
Collecting customtkinter
  Using cached customtkinter-5.2.2-py3-none-any.whl.metadata (677 bytes)
Collecting darkdetect (from customtkinter)
  Using cached darkdetect-0.8.0-py3-none-any.whl.metadata (3.6 kB)
Collecting packaging (from customtkinter)
  Downloading packaging-24.2-py3-none-any.whl.metadata (3.2 kB)
Using cached customtkinter-5.2.2-py3-none-any.whl (296 kB)
Using cached darkdetect-0.8.0-py3-none-any.whl (9.0 kB)
Downloading packaging-24.2-py3-none-any.whl (65 kB)
Installing collected packages: packaging, darkdetect, customtkinter
Successfully installed customtkinter-5.2.2 darkdetect-0.8.0 packaging-24.2

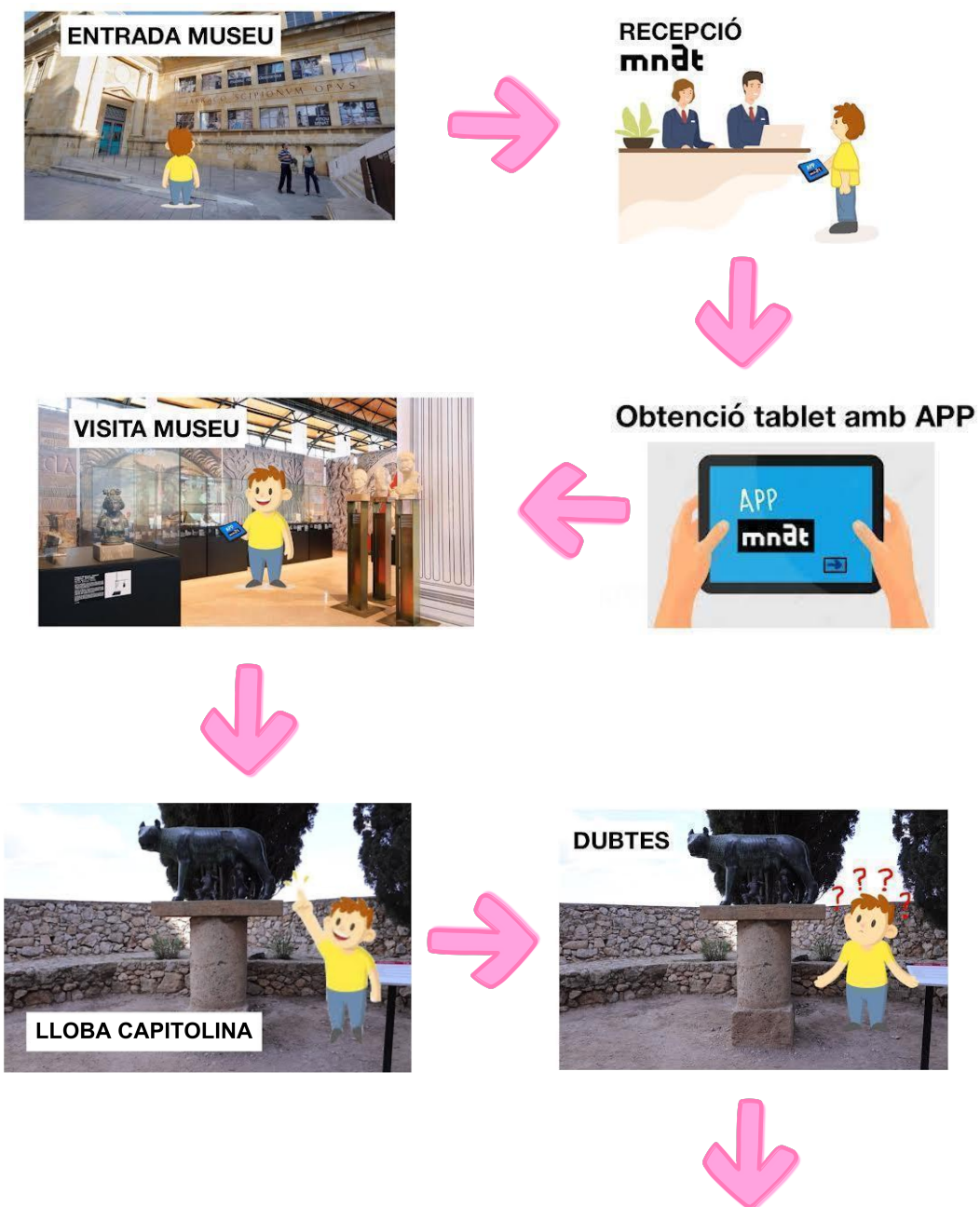
[notice] A new release of pip is available: 24.2 -> 24.3.1
[notice] To update, run: C:\Users\laura\AppData\Local\Programs\Python\Python312\python.exe -m pip install --upgrade pip
```

Imatge 15: Instal·lació d'una llibreria de Python

4.2. APLICACIÓ D'ESCRITORI

4.2.1. ÚS DE L'APLICACIÓ D'ESCRITORI AL MUSEU

A la següent infografia mostro com seria una visita guiada pel museu amb l'ajuda de la intel·ligència artificial





Imatge 16: Representació d'una visita guiada per IA al MNAT

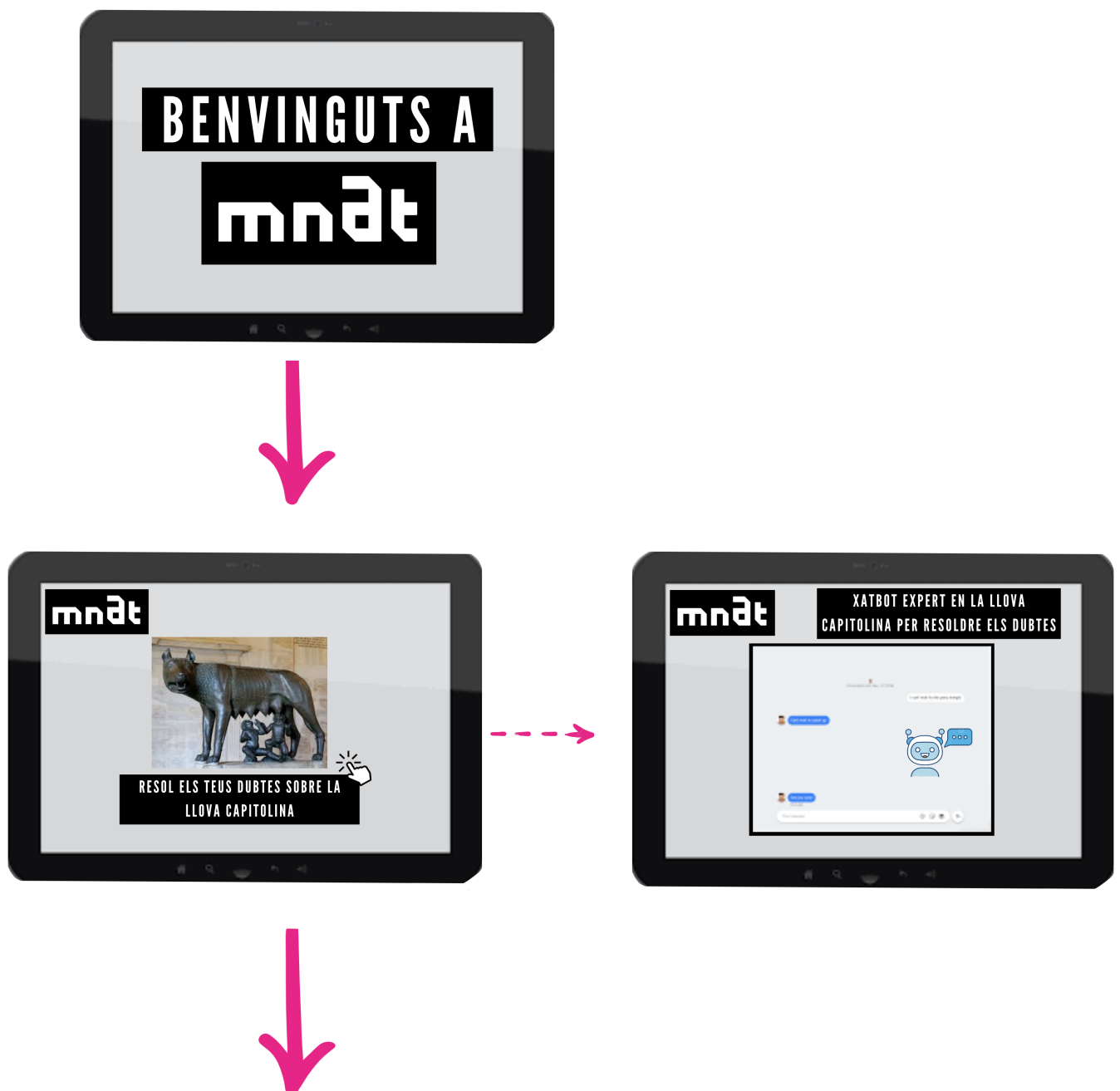
Primer de tot, quan el visitant entra al museu, se li entrega una tauleta tàctil amb l'aplicació d'escriptori. Mentre que l'usuari va passejant pel museu pot anar mirant les obres i quan li sorgeix algun dubte o vol obtenir més informació, només ha d'obrir el xatbot de l'aplicació i preguntar.

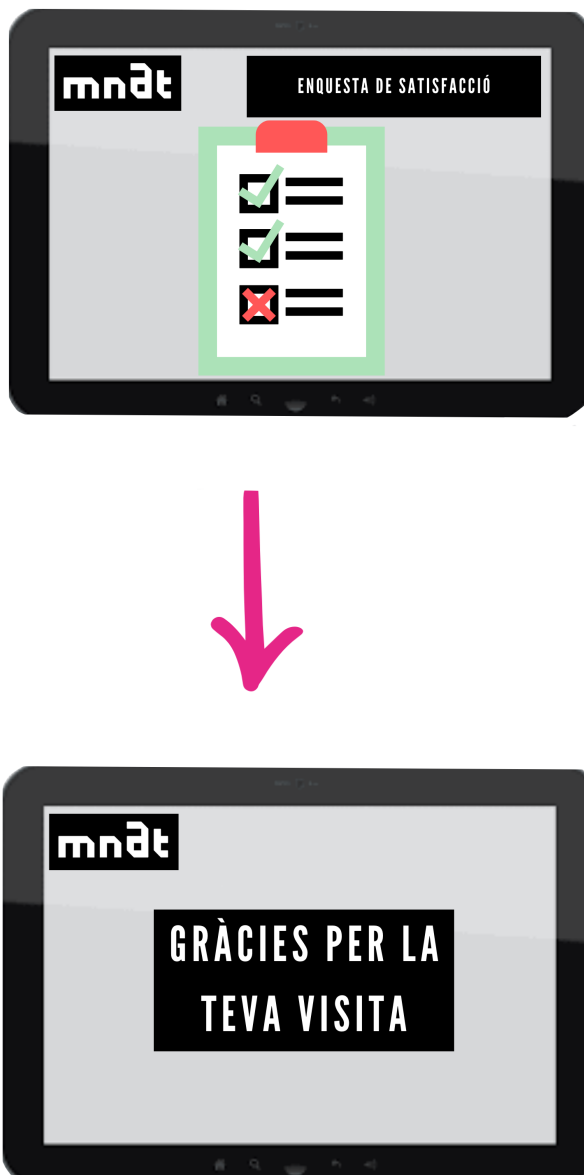
En acabar la visita, l'usuari pot respondre una enquesta de satisfacció per valorar la seva experiència al museu.

4.2.2. FLUX DE NAVEGACIÓ

La següent infografia mostra el flux de pantalles de l'aplicació d'escriptori dissenyada per fer visites guiades al MNAT.

L'aplicació s'obre amb una pàgina de benvinguda al MNAT. Aquesta pàgina conté un botó que, quan es prem, amaga la finestra inicial i crea la finestra següent amb unes mides concretes perquè s'adapti a la pantalla del meu ordinador.





Imatge 17: Flux de pantalles de l'aplicació

La segona finestra conté una fotografia de l'obra d'art, la Lloba Capitolina, i un botó que quan es prem fa que s'obri una finestra on l'usuari pot interactuar amb el xatbot resolent els seus dubtes. Aquesta finestra conté una zona on l'usuari pot introduir les preguntes, un botó per enviar-les i una capsa de diàleg per veure la conversa amb el xatbot.

Per continuar, hi ha un botó per avançar de pantalla on hi ha una enquesta de satisfacció perquè l'usuari valori diferents punts de la seva visita al museu.

Finalment, hi ha un botó per anar a l'última pàgina de l'aplicació d'escriptori on es diu adéu al visitant, donant-li les gràcies.

4.2.3. CREACIÓ DE PANTALLES

En les diferents pantalles del flux de navegació que he explicat en l'apartat anterior, he creat una sèrie d'etiquetes, camps d'entrada, botons, imatges, radiobuttons, àrees de text i barres de desplaçament. Per implementar tots aquests widgets he utilitzat la llibreria CUSTOMTKINTER que permet crear interfícies gràfiques d'usuari (GUI) amb un look&feel més modern i personalitzat que no pas la llibreria TKINTER.



Imatge 18: Primera pantalla de l'aplicació

A la primera finestra, amb el codi `customtkinter.CTkImage()` vaig posar la imatge de benvinguda al MNAT i vaig ajustar la mida amb la funció `size()` perquè ocupi tota la pantalla de la tauleta.



Imatge 19: Segona pantalla de l'aplicació

En segon lloc, es presenta la finestra on es troben les imatges de les obres d'art. Per posar aquestes imatges també vaig utilitzar el codi `customtkinter.CTkImage()` i la funció `size()` perquè les imatges estiguin justament a dalt dels botons que obrin la finestra del xatbot.



Imatge 20: Tercera pantalla de l'aplicació

En tercer lloc, tenim la pantalla on es convida l'usuari a fer l'enquesta de valoració on també vaig utilitzar el `customtkinter.CTkImage()` per posar la imatge de fons i amb la funció `size()` la vaig ajustar perquè ocupés tot l'espai.

Imatge 21: Quarta pantalla de l'aplicació

En quart lloc, vaig utilitzar una sèrie de widgets perquè els visitants poguessin introduir la seva opinió.

Per posar les preguntes vaig fer servir el codi `customtkinter.CTkLabel()`, especificant la lletra i la posició on es troba la pregunta dins la pantalla.

Per posar els radiobuttons vaig utilitzar el codi `customtkinter.CTkRadioButton()`. Després, amb `customtkinter.CTkSlider()` vaig fer unes barres de desplaçament configurades perquè es desplacin del 0 al 100, en horitzontal i de 10 en 10.

I finalment, perquè l'usuari pugui respondre una pregunta escrivint, vaig utilitzar `customtkinter.CTkTextbox()` per crear una àrea de text, indicant la seva alçada i amplada. Per gestionar les respostes introduïdes per l'usuari dins l'àrea de text, vaig posar dos botons. El primer botó conté una funció "delete" que quan es prem, esborra el text introduït. El segon botó té una funció "enviar", que en clicar-lo, surt un missatge a la pantalla assegurant que s'ha desat la resposta.



Imatge 22: Cinquena pantalla de l'aplicació

En l'última pantalla, igual que en les altres, per posar la imatge de fons vaig utilitzar `customtkinter.CTkImage()` i la funció `size()` per fer l'ajustament. També vaig posar un botó que tanca la finestra i torna a la pantalla inicial de benvinguda.

4.3. XATBOT

Abans de començar a explicar aquest apartat, vull aclarir que el meu xatbot està basat en l'article [13] de la webgrafia. En aquest article s'explica com construir un xatbot amb intel·ligència artificial pas a pas. Vaig llegir-lo detingudament i vaig implementar el codi que hi ha escrit, fent petites modificacions perquè algunes parts del codi, tal com estan a l'article, em donaven error a l'hora de fer l'execució.

4.3.1. FITXER DE DADES

Tal com he explicat a l'apartat 3.4.3, els xatbots utilitzen una base de dades on s'emmagatzema tota la informació necessària perquè la IA pugui respondre, de la millor manera possible, els dubtes de l'usuari. Aquesta base de dades conté

les qüestions que pot fer l'usuari i les respostes de la intel·ligència artificial, com preguntes freqüents i també salutacions i acomiadaments que defineixen els diferents camins que pot seguir una conversa basada en les entrades de l'usuari. En el meu cas, la base de dades és un fitxer JSON, que es pot veure tot complet en l'annex 7.1.

El xatbot utilitza la informació del fitxer JSON per processar les entrades dels usuaris i generar respostes adequades. Això facilita una gestió eficient de la informació, així com l'actualització i el manteniment del sistema.

```
{
  "intents": [
    {
      "tag": "salutacions",
      "patterns": ["Hola", "Com estas", "Hi ha algu?", "Bon dia", "Que tal?", "Bona tarda"],
      "responses": ["Hola, gracies per preguntar", "Un plaer tornar-te a veure", "Hola, com puc ajudar?"],
      "context": [""]
    },
    {
      "tag": "comiat",
      "patterns": ["Adeu", "Fins despres", "Fins una atra", "Un plaer xerrar amb tu, adeu", "Fins la propera"],
      "responses": ["Ens veiem!", "Que tinguis un bon dia", "Adeu! Torna aviat"],
      "context": [""]
    },
    {
      "tag": "noresposta",
      "patterns": [],
      "responses": ["Ho sento, no t'entenc", "Si us plau, dona'm més informació", "No estic segur d'entendre't"],
      "context": [""]
    }
  ]
}
```

Imatge 23: Algunes etiquetes i patrons de preguntes del fitxer JSON

4.3.2. PREPROCESSAMENT DE DADES

Per fer un machine learning (ML) o un deep learning (DL) es transformen les dades del fitxer JSON mitjançant les operacions que indico a continuació:

- El primer pas és la creació de tokens a partir de les dades del fitxer. La tokenització, tal com he dit en l'apartat 3.2.1, és un procés que consisteix a dividir les frases en petites parts, com ara paraules. Per fer la tokenització s'utilitza la funció `nltk.word_tokenize()`. Es creen dues llistes, una de paraules i l'altra d'etiquetes.
- A continuació, es lematitza cada paraula de la llista i s'eliminen paraules duplicades. Com he explicat a l'apartat 3.2.2, la lematització és el procés de convertir una paraula en la seva arrel.

- Finalment, es creen un parell d'arxius pickle per emmagatzemar els objectes de Python (paraules i classes) que s'utilitzaran per fer les prediccions. Les classes són les etiquetes.

4.3.3. CREACIÓ DE DADES D'ENTRENAMENT

Per entrenar la IA i fer que sigui capaç de prendre decisions amb una bona precisió s'han de crear dades d'entrenament, proporcionant l'entrada i la sortida.

L'entrada és el patró (possibles preguntes) i la sortida la classe a la qual pertany el patró d'entrada, és a dir, l'etiqueta.

Com que la IA està basada en un conjunt d'algorismes matemàtics, es fa una transformació numèrica de les dades.

4.3.4. CONSTRUCCIÓ DEL MODEL

Un cop creades les dades d'entrenament, s'utilitza la funció Sequential() de la llibreria Keras per construir un model basat en una xarxa neuronal profunda de 3 capes.

Tot seguit, s'entrena el model amb la funció model.fit(), realitzant 200 iteracions. Això permet aconseguir un model amb una precisió del 100%. El xatbot serà capaç de predir totes les respostes i relacionar les entrades per donar sortides.

```
Epoch 194/200
0[1m 1/110[0m 0[32m -----0[0m 0[1m0s[0m 28ms/step - accuracy: 0.8000 - loss: 0.6100
0[0m[37m[0m 0[1m0s[0m 1ms/step - accuracy: 0.9211 - loss: 0.2385
Epoch 195/200
0[1m 1/110[0m 0[32m -----0[0m 0[1m0s[0m 31ms/step - accuracy: 1.0000 - loss: 0.1042
0[0m[37m[0m 0[1m0s[0m 1ms/step - accuracy: 0.9527 - loss: 0.1186
Epoch 196/200
0[1m 1/110[0m 0[32m -----0[0m 0[1m0s[0m 25ms/step - accuracy: 1.0000 - loss: 0.0026
0[0m[37m[0m 0[1m0s[0m 2ms/step - accuracy: 0.9080 - loss: 0.1535
Epoch 197/200
0[1m 1/110[0m 0[32m -----0[0m 0[1m0s[0m 16ms/step - accuracy: 0.8000 - loss: 0.2210
0[0m[37m[0m 0[1m0s[0m 2ms/step - accuracy: 0.8802 - loss: 0.2125
Epoch 198/200
0[1m 1/110[0m 0[32m -----0[0m 0[1m0s[0m 41ms/step - accuracy: 0.8000 - loss: 0.3357
0[0m[37m[0m 0[1m0s[0m 687us/step - accuracy: 0.9345 - loss: 0.1404
Epoch 199/200
0[1m 1/110[0m 0[32m -----0[0m 0[1m0s[0m 32ms/step - accuracy: 1.0000 - loss: 9.8439e-04
0[0m[37m[0m 0[1m0s[0m 2ms/step - accuracy: 0.9393 - loss: 0.1330
Epoch 200/200
0[1m 1/110[0m 0[32m -----0[0m 0[1m0s[0m 33ms/step - accuracy: 1.0000 - loss: 0.0044
0[0m[37m[0m 0[1m0s[0m 1ms/step - accuracy: 0.9185 - loss: 0.1232
```

Imatge 24: Últimes iteracions de l'entrenament

El model creat i entrenat es guarda en un fitxer anomenat “chatbot_model.h5”.

Model: "sequential"

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 128)	14,336
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 64)	8,256
dropout_1 (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 14)	910

Total params: 47,006 (183.62 KB)
Trainable params: 23,502 (91.80 KB)
Non-trainable params: 0 (0.00 B)
Optimizer params: 23,504 (91.82 KB)
None
model created

Imatge 25: Creació del model d'intel·ligència artificial

4.3.5. PREDICCIÓ DE RESPOSTES

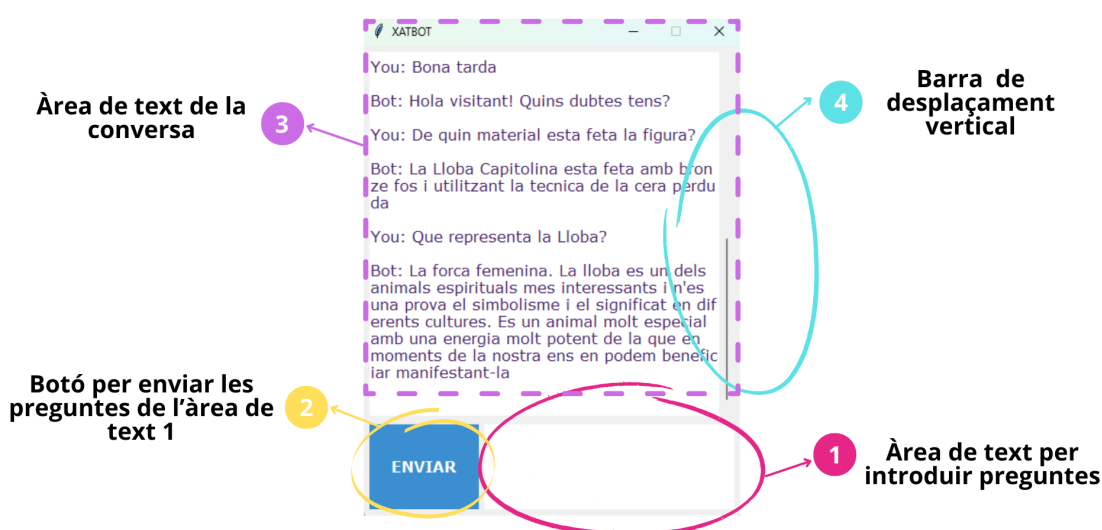
Per poder predir la classe o etiqueta de les preguntes fetes per l'usuari i obtenir una resposta per a cada pregunta, dintre del codi de l'aplicació del museu es carrega el model entrenat de la intel·ligència artificial mitjançant la instrucció `model = load_model('chatbot_model.h5')`.

Després, es carrega l'arxiu JSON i els arxius pickle de paraules i classes generats durant la creació del model.

Tal com es pot veure al codi de l'annex 7.3, dintre de la funció `lloba_capitolina()`, quan l'usuari fa una pregunta al xatbot, es crida a la funció `chatbot_response()` passant-li el text de la qüestió com a paràmetre d'entrada. Aquesta funció utilitza el model per determinar la classe o etiqueta que correspon a la pregunta. Això ho fa cridant a la funció `predict_class()`. Llavors, crida a la funció `getResponse()` per obtenir de manera aleatòria una de les respostes associades a la classe obtinguda a partir del fitxer de dades JSON.

4.3.6. FINESTRA DEL XATBOT

Per crear la finestra del xat on l'usuari i la IA puguin interactuar, s'utilitzen les funcions de la llibreria Tkinter. La finestra tindrà una dimensió de 400 (píxels horitzontals) x 500 (píxels verticals), hi haurà una àrea de text on el visitant podrà introduir els seus dubtes, un botó per poder enviar la pregunta al xatbot, una àrea on apareixerà el text de la conversa i, finalment, una barra de desplaçament vertical per poder veure totes les preguntes i respostes.



Imatge 26: Esquema finestra del xatbot

4.3.7. INTEGRACIÓ DEL XATBOT A L'APLICACIÓ D'ESCRITORI

Perquè la finestra creada anteriorment es pugui utilitzar a l'aplicació del MNAT, dintre del codi de l'aplicació d'escriptori vaig fer una funció anomenada `lloba_capitolina()`. Quan es prem el botó que hi ha a sota de la foto de l'obra d'art, es crida a la funció i s'obre la finestra amb el xatbot. Aquest codi es pot veure a l'annex 7.3.

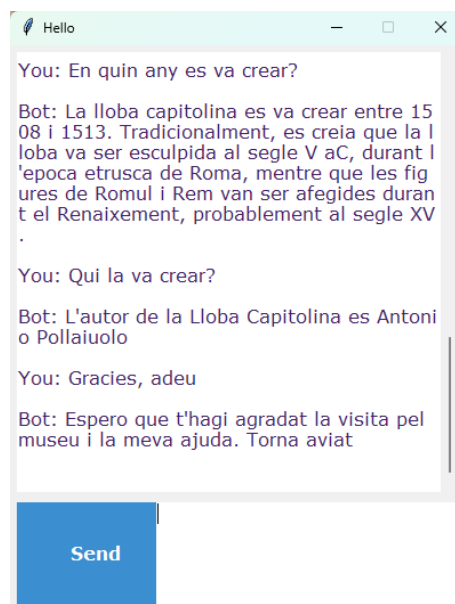
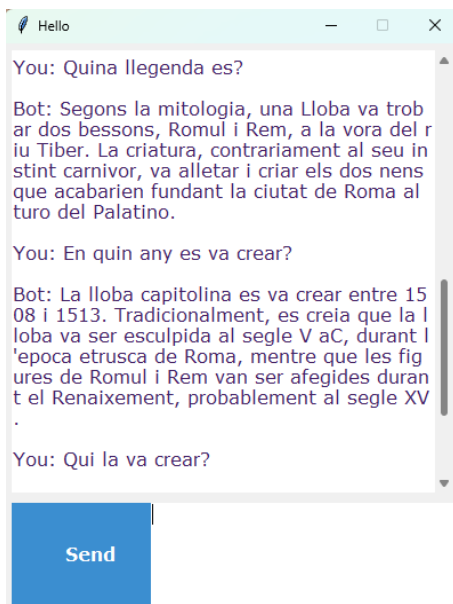
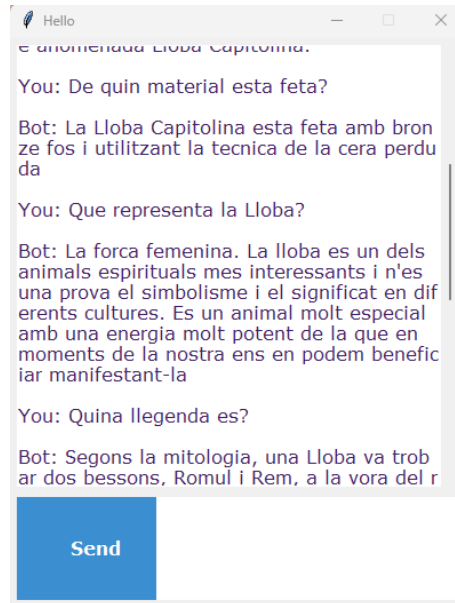
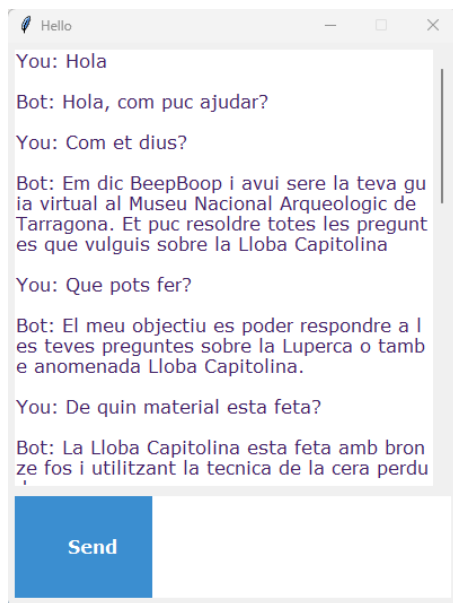


Imatge 27: Finestra del xatbot dins l'aplicació del MNAT

4.3.8. EXECUCIÓ DEL XATBOT

Per comprovar que el xatbot és capaç d'interactuar correctament amb els usuaris i respondre les seves preguntes sobre la Lloba Capitolina, he realitzat un seguit de proves.

A continuació, mostro captures de pantalla de la finestra del xatbot amb una conversa mantinguda amb un usuari. Tal com es pot veure, funciona bé.



Imatge 28: Proves de funcionament del xatbot

5. CONCLUSIONS

Abans de començar aquest projecte tenia una visió molt reduïda sobre la intel·ligència artificial. No tenia cap idea de com funcionava i, com la majoria de les persones, pensava que la IA només era el ChatGPT i els robots. Però després d'haver acabat el meu TdR, puc afirmar que la IA no és només això, ja que he investigat que és el machine learning, el deep learning i he construït una IA basada en xarxes neuronals per a un xatbot. Ara tinc una visió molt més àmplia d'aquest camp perquè he creat el cervell de la meua IA (fitxer de dades JSON), he creat dades d'entrenament, he entrenat a la IA, he creat el model entrenat i l'he utilitzat per predir etiquetes o classes per tal de donar les respostes més adequades als dubtes de l'usuari.

Aquest treball m'ha resultat molt útil per poder complir el meu objectiu de tenir un contacte més profund amb la intel·ligència artificial i la programació, i poder posar en pràctica alguns coneixements que vaig adquirir el curs passat a l'assignatura de programació, amb l'ajuda de la meua professora i guia en el meu TdR.

Gràcies a la recerca d'informació i a les nombroses proves que he fet, he aconseguit introduir-me amb més detall al llenguatge de programació Python, aprenent, a més a més, el funcionament de la IA. Així mateix, he fet ús de molts diagrames, taules de classificació, imatges i croquis, que m'han permès adquirir més destresa en la competència digital, cosa que em serà molt útil en un futur.

Com en cada repte, han sorgit moltes dificultats i problemes a mesura que he anat avançant, però m'ha ajudat molt el fet d'haver plantejat uns objectius que volia complir en el meu treball, els objectius descrits en l'apartat 1.2. Tot i això, he hagut de fer modificacions a la idea inicial perquè vaig haver de descartar algunes idees i implementar nous elements per tal d'afavorir el resultat.

He tingut una experiència molt enriquidora fent els diferents apartats de què consta aquest treball. Tots han plantejat noves qüestions que he hagut de resoldre i, a la vegada, cadascun d'ells ha estat una peça essencial per culminar el producte final. Tot el temps que he dedicat a fer aquest projecte

m'ha permès adquirir nous coneixements en el camp de la intel·ligència artificial i la programació.

Per finalitzar el meu treball de recerca, mostraré si he assolit els quatre objectius que em vaig proposar al començament, quins problemes he tingut, com els he pogut solucionar, i quins són els punts de millora del meu producte final.

5.1. ASSOLIMENT D'OBJECTIUS



He pogut assolir perfectament el primer objectiu, ja que he fet servir l'aprenentatge automàtic o machine learning per fer un xatbot. Però primer de tot vaig haver de buscar informació sobre què és una IA perquè no tenia gaires coneixements sobre això.



Considero que he après profundament sobre els assistents virtuals. He pogut programar-ne un, específicament, perquè estigui inspirat en el Museu Nacional Arqueològic de Tarragona.



En tercer lloc, he pogut dissenyar i construir l'aplicació d'escriptori basada en el MNAT, crear el xatbot i integrar-lo en l'aplicació.



Finalment, com no podia fer un xatbot per cada obra d'art perquè això hagués requerit molt de temps de creació, més del temps disponible per fer el TdR, vaig especialitzar-me en la Lloba Capitolina. Per tant, el meu bot és expert en aquesta obra d'art i és capaç de respondre els diferents dubtes que pugui tenir l'usuari al respecte.

Així doncs, puc afirmar amb seguretat que he assolit tots els objectius que m'havia proposat en aquest treball.

També he pogut demostrar que és possible introduir un assistent virtual al MNAT per ajudar els visitants del museu en el seu recorregut, contestant així la pregunta de la hipòtesi de l'apartat 1.2 “es podria introduir un assistent virtual al MNAT per ajudar els visitants del museu en el seu recorregut?”.

A més a més, volia destacar una sèrie d'aspectes importants sobre el meu producte final:

- Per fer que l'aplicació d'escriptori del MNAT sigui el més atractiva possible, vaig haver de crear tots els fons de les finestres amb CANVA. De fet, el disseny gràfic és un punt fonamental en la construcció d'aplicacions.
- Em va resultar complicat fer que tots els elements visibles i codificats de l'aplicació quedessin ben ajustats a les finestres. Per aconseguir-ho, vaig haver de fer moltes modificacions en la ubicació dels botons, les imatges, els widgets de l'enquesta i la mida de la lletra.
- Un altre aspecte rellevant és que vaig haver d'aprendre a utilitzar la llibreria CustomTkinter. Fins ara només havia treballat amb la llibreria Tkinter, que fa servir un disseny més bàsic i avorrit. Gràcies a aprendre aquesta nova llibreria, el disseny de tots els widgets són més moderns i estètics.
- També vaig haver d'aprendre a fer servir altres llibreries per crear la IA, com ara la llibreria KERAS.
- A més, vaig haver de buscar informació per poder fer la navegació entre finestres, que no és senzill, ja que implica crear funcions anidades (una funció està dintre de l'altra) dins del codi de l'aplicació.

En un inici em vaig plantejar fer el xatbot amb una llibreria que es diu CHATTERBOT, però després d'investigar més sobre ella, vaig veure que estava obsoleta i que, per tant, no es podia fer servir en el meu treball. Per aquest motiu vaig haver de buscar una nova solució. Afortunadament, vaig trobar l'article [13] de la webgrafia, on hi ha un codi per construir un xatbot i la seva finestra, que em va ser més útil i fàcil d'adaptar al meu treball.

5.2. PROBLEMES I SOLUCIONS

Però com tot, sempre hi ha obstacles que s'han de poder solucionar per arribar a la meta final. Al llarg de tot el procés de disseny i construcció m'he anat trobant amb diverses dificultats que he hagut de resoldre o buscar alternatives per a obtenir el producte final.

Primer de tot, quan vaig crear el fitxer de dades JSON, vaig posar accents i quan es mostraven en l'àrea de la conversa a la finestra de l'assistent virtual, els accents es transformaven en caràcters estranys. Llavors, com no vaig trobar cap solució viable, vaig treure tots els accents del fitxer per facilitar la lectura. Podria ser que aquest problema sigui degut a un tema de codificació UNICODE, que s'hauria d'investigar amb molt deteniment. És per això, que el meu xatbot, no utilitza accents.

El segon problema que vaig tenir va ser que al meu ordinador personal tenia particions del disc dur i em vaig quedar sense espai d'emmagatzematge en totes elles. Això va fer que no pogués programar ni guardar més codi de l'aplicació d'escriptori del MNAT. Per solucionar-ho vaig traslladar el programa i totes les imatges a una memòria USB per poder-ho conservar i no perdre res del que havia fet.

En tercer lloc, vaig intentar instal·lar-ho tot al meu ordinador de l'institut i obrir el codi des de la memòria USB. Però a l'ordinador de l'institut no podia executar el codi, ja que em donaven problemes les llibreries. I alhora, la memòria USB on vaig traslladar el treball era defectuosa perquè era específicament per a emmagatzemar imatges. Llavors, el codi dintre de l'USB es va esborrar i ja no podia accedir al codi del meu ordinador personal perquè havia deixat de funcionar totalment.

Per sort, mesos abans li havia enviat un correu a la meva tutora del TdR amb el codi de l'aplicació d'escriptori, i gràcies a això vaig poder recuperar part del meu treball. I també, el meu pare va aconseguir arreglar el problema d'emmagatzematge que tenia en el meu ordinador personal i vaig poder tornar a fer el treball com al començament. Però un inconvenient va ser que l'ordinador estava buit sense cap informació i vaig haver de tornar a

instal·lar-ho tot, l'IDLE i les llibreries, una altra vegada.

El quart problema és que em vaig adonar que al codi del correu hi faltava una cosa: la finestra del xatbot, ja que es tractava d'una versió de codi més antiga. Vaig anar a la carpeta compartida del Drive amb la meva tutora del TdR perquè allí vam fer una còpia del codi que hi havia a la memòria USB, però vaig trobar-me que també estava buida. Llavors, per solucionar-ho vaig haver de refer el codi de la finestra partint des de zero. Per això, vaig tornar a codificar la funció `lloba_capitolina()` que inclou la finestra del xatbot, però també la predicció de les respostes per contestar les preguntes de l'usuari i la integració de la finestra a l'aplicació. Tot això em va comportar més temps perquè, tot i que ho havia fet un cop, aquesta part del codi és més difícil i per aquest motiu vaig haver de tornar a mirar l'article del xatbot (número [13] de la webgrafia) per refer el codi i no deixar-me res.

En acabar de programar el codi que havia perdut, el vaig enviar per correu a la meva tutora del treball per evitar de nou tots aquests problemes i tenir l'última versió del codi en un lloc segur.

En últim lloc, em vaig trobar que les llibreries de la IA (TENSORFLOW i KERAS) no estaven actualitzades per a la versió de Python que vaig instal·lar al meu ordinador personal després de recuperar-lo (versió 3.13 d'IDLE). Per tant, vaig haver de fer un "downgrade", és a dir, instal·lar una versió anterior (la 3.12). D'aquesta manera ja vaig poder instal·lar correctament totes les llibreries.

Finalment, i després de resoldre tot això, vaig poder acabar la meva aplicació d'escriptori i el xatbot. Gràcies a tots aquests problemes, he après per futurs projectes que és molt important tenir còpies de seguretat guardades en diferents llocs. Així evitarem perdre una informació molt valuosa i en la qual hem destinat moltes hores de treball.

5.3. PUNTS DE MILLORA

Tal com vaig comentar en la introducció d'aquest treball, la meva aplicació i el meu xatbot són un prototip, per tant, encara hi falten coses per implementar i punts per millorar:

- S'hauria d'investigar el problema dels accents del xatbot. Com he dit abans, si poso accents a les paraules del fitxer JSON, els accents es transformen en caràcters estranys i això dificulta la lectura de les respostes del xatbot dintre de la conversa mantinguda amb l'usuari. Una possible manera de solucionar-ho seria canviar la codificació, utilitzant caràcters UNICODE en lloc de caràcters ASCII en la base de dades JSON.
- Un altre punt de millora és que només he pogut crear un bot expert en la Lloba Capitolina, ja que fer tants xatbots com obres d'art hi ha al museu, suposaria molt de temps. Al cap i a la fi, el meu treball és un prototip i fer tots els xatbots del MNAT seria un projecte d'empresa amb un equip de treball format per diversos informàtics i amb molts mesos de creació. A més, al museu, fer aquest projecte li costaria diners.
- En tercer lloc, les dades que s'introdueixen en l'enquesta per valorar la visita al MNAT no s'emmagatzemen en cap lloc. Per tant, aquestes dades es podrien emmagatzemar en un fitxer TXT o una base de dades, com ara SQLite, que forma part de Python i no s'ha d'instal·lar cap llibreria. Així es podrien fer servir les valoracions per poder analitzar-les, treure conclusions, millorar l'aplicació i l'experiència al museu dels visitants.
- El quart punt de millora és que es podria guardar, en una base de dades, totes les converses entre els usuaris i la IA del xatbot per veure quines són les preguntes que li fan i, per tant, modificar el fitxer JSON, incloent-hi noves etiquetes (classes), preguntes i respostes. Així es podria fer una millora contínua del fitxer JSON actualitzant-lo segons les converses mantingudes.

- Un aspecte rellevant per millorar és que quan l'usuari pregunta sobre un altre tema que no té res a veure amb la Lloba Capitolina i que no es troba tipificat al fitxer JSON, el bot contesta coses relacionades amb aquesta obra d'art i no respon a la qüestió realitzada. En aquest cas, s'hauria de fer alguna modificació al fitxer JSON o al codi perquè en aquestes situacions la IA pugui respondre alguna frase com ara "No dispo de informació per contestar la pregunta realitzada". Vaig estar investigant per trobar la forma de solucionar-ho, però no és fàcil de fer i no he trobat la solució encara.
- L'últim punt important és que per poder posar en marxa al museu l'aplicació que he desenvolupat i el xatbot de la Lloba Capitolina, prèviament s'haurien de fer moltes proves amb diferents usuaris per definir, el millor possible, les preguntes i les respostes del cervell del xatbot.

6. BIBLIOGRAFIA I WEBGRAFIA

- [1] Raschka, S., Mirjalili, V. (2019). *Python Machine Learning*. Editorial Marcombo.
- [2] Blog Social Media & SEO (2024). *Frases intel·ligència artificial IA*.
<https://www.manuduque.com/frases-inteligencia-artificial-ia/>
- [3] Naciones Unidas (s/f). *Objetivos de Desarrollo Sostenible*.
<https://www.un.org/sustainabledevelopment/es/sustainable-development-goals/>
- [4] Plan de recuperación. (s/f). *Qué es la inteligencia artificial*.
<https://planderecuperacion.gob.es/noticias/que-es-inteligencia-artificial-ia-prtr>
- [5] Europarl (08/09/2020). *¿Qué es la inteligencia artificial?*
<https://www.europarl.europa.eu/topics/es/article/20200827STO85804/que-es-la-inteligencia-artificial-y-como-se-usa>
- [6] Machine learning (09/03/2023). *Machine learning - aprenentatges*.
<https://ml.araesdema.cat/aprenentatges#h.yfcjxw402h7y>
- [7] HubSpot (s/f). *Tipos de inteligencia artificial que puedes usar este 2024*.
<https://blog.hubspot.es/marketing/tipos-inteligencia-artificial#tipos>
- [8] IBM (17 de junio de 2024). *¿Qué es el deep learning?*
<https://www.ibm.com/es-es/topics/deep-learning>
- [9] IBM (s/f). *¿Qué son las redes neuronales?*
<https://www.ibm.com/es-es/topics/neural-networks>
- [10] IBM (s/f). *¿Qué es el PLN (procesamiento de lenguaje natural)?*
<https://www.ibm.com/es-es/topics/natural-language-processing>
- [11] Decide (12/09/2021). *Procesamiento del Lenguaje Natural (PLN o NLP): qué es y para qué se utiliza*.

<https://decidesoluciones.es/procesamiento-del-lenguaje-natural-pln-o-nlp-que-es-y-para-que-se-utiliza/>

[12] QServus (09/02/2024). *Procesamiento del lenguaje natural (PNL): 5 técnicas clave.*

<https://blog.qservus.com/procesamiento-del-lenguaje-natural-pln-5-tecnicas-clave/#:~:text=La%20lematizaci%C3%B3n%20va%20un%20paso.Caminar%2C%20caminaba%2C%20caminaron%2C%20caminemos>

[13] Medium (20/04/2021). *Tutorial: Construye tu primer chatbot usando NLTK y Keras.*

<https://hiteshmishra708.medium.com/tutorial-chatbot-usando-nltk-keras-d1f8066e404c>

[14] IBM (s/f). *¿Qué es un chatbot?*

<https://www.ibm.com/es-es/topics/chatbots>

[15] Freshchat (s/f). *Tipos de chatbot.*

<https://www.freshworks.com/es/live-chat-software/chatbots/three-types-of-chatbots/>

[16] AWS (s/f). *¿Qué es Python?*

<https://aws.amazon.com/es/what-is/python/>

[17] Python.org (07/12/2024). *El tutorial de Python.*

<https://docs.python.org/es/3/tutorial/>

[18] HubSpot (28/03/2024). *Qué es Python: para qué sirve y funciones (+ recursos).*

<https://blog.hubspot.es/website/que-es-python#que-es>

[19] Python.org (07/12/2024). *IDLE - documentación de Python - 3.13.1.*

<https://docs.python.org/es/3/library/idle.html>

[20] Inába (s/f). *Qué es IDLE: El entorno de programación de Python.*

<https://www.inabaweb.com/que-es-idle-el-entorno-de-programacion-de-python/>

7. ANNEXOS

7.I. FITXER DE DADES DEL XATBOT

```
{
  "intents": [
    {
      "tag": "salutations",
      "patterns": ["Hola", "Com estas", "Hi ha algu?", "Bon dia", "Que tal?", "Bona tarda", "Bon dia", "hola", "hello", "Bona nit", "Ei!"],
      "responses": ["Hola, gracies per parlar amb mi. Et puc ajudar en algo?", "Hola visitant! Quins dubtes tens?", "Hola, com puc ajudar?", "Hola! Que necessites?."],
      "context": [""]
    },
    {
      "tag": "funcions",
      "patterns": ["Que fas?", "Quina es la teva funcio?", "Que pots fer?", "En que em pots ajudar?", "Quin es el teu treball?", "Com em pots ajudar?"],
      "responses": ["El meu objectiu es poder respondre a les teves preguntes sobre la Luperca o tambe anomenada Lloba Capitolina.", "Doncs soc una IA experta en l'obra d'art de la Lloba Capitolina.", "Puc donar-te solucions a totes les teves preguntes sobre la Lloba Capitolina."],
      "context": [""]
    },
    {
      "tag": "comiat",
      "patterns": ["Adeu", "Fins despres", "Fins una atra", "Un plaer xerrar amb tu, adeu", "Fins la propera"],
      "responses": ["Ha sigut un plaer parlar amb tu. Ens veiem!", "Espero poder parlar amb tu un altre cop. Que tinguis un bon dia", "Gracies per la teva visita, ha estat un plaer poder resoldre els teus dubtes. Adeu!", "Espero que t'hagi agradat la visita pel museu i la meva ajuda. Torna aviat"],
      "context": [""]
    },
    {
      "tag": "noresposta",
      "patterns": [],
      "responses": ["No sento, no t'entenc", "Si us plau, dona'm més informacio", "No estic segur d'entendre't"],
      "context": [""]
    },
    {
      "tag": "presentacio",
      "patterns": ["Com et dius?", "Quin es el teu nom?", "Et dius?", "Em pots dir el teu nom?", "Tens algun nom?"],
      "responses": ["Em dic BeepBoop i avui sere la teva guia virtual al Museu Nacional Arqueologic de Tarragona. Et puc resoldre totes les preguntes que vulguis sobre la Lloba Capitolina.", "El meu nom es BeepBoop i avui t'ajudare a resoldre tots els dubtes de la Lloba Capitolina al Museu Nacional Arqueologic de Tarragona"],
      "context": [""]
    },
    {
      "tag": "Materials i tecnica",
      "patterns": ["De quin material esta feta la figura?", "Quina tecnica s'utilitza a la figura?", "Com es la tecnica que van utilitzar per fer-la?"],
      "responses": ["Segons la mitologia, una Lloba va trobar dos bessons, Romul i Rem, a la vora del riu Tiber. La criatura, contrariament al seu instint carnivor, va alleitar i criar els dos nens que acabarien fundant la ciutat de Roma al turo del Palatino."],
      "context": [""]
    },
    {
      "tag": "Llegenda de la estatua",
      "patterns": ["Quina es la llegenda de la figura?", "Com es la llegenda de la figura?", "Que diu la llegenda?", "Que representa la estatua?", ""],
      "responses": ["Segons la mitologia, una Lloba va trobar dos bessons, Romul i Rem, a la vora del riu Tiber. La criatura, contrariament al seu instint carnivor, va alleitar i criar els dos nens que acabarien fundant la ciutat de Roma al turo del Palatino."],
      "context": [""]
    },
    {
      "tag": "Any de creacio de la estatua",
      "patterns": ["Quan es va fer la Lloba Capitolina?", "En quin any es va crear la estauta?", "Quin va ser l'any de creacio?", "?", ""],
      "responses": ["La lloba capitolina es va crear entre 1508 i 1513. Tradicionalment, es creia que la lloba va ser esculpida al segle V aC, durant l'epoca etrusca de Roma, mentre que les figures de Romul i Rem van ser afegides durant el Renaixement, probablement al segle XV."],
      "context": [""]
    },
    {
      "tag": "Autor de la estatua",
      "patterns": ["Qui va ser l'autor de la Lloba Capitolina?", "Qui va crear la luperca?", "Com es diu l'autor de la Lloba Capitolina?", "", ""],
      "responses": ["L'autor de la Lloba Capitolina es Antonio Follaiuolo"],
      "context": [""]
    },
    {
      "tag": "Que representa la luperca",
      "patterns": ["Que representa la luperca?", "Quina representacio te la lloba capitolina?", "Que significa la lloba?"],
      "responses": ["La forca femenina. La lloba es un dels animals espirituals mes interessants i n'es una prova el simbolisme i el significat en diferents cultures. Es un animal molt especial amb una energia molt potent de la que en moments de la nostra ens en podem beneficiar manifestant-la"],
      "context": [""]
    },
    {
      "tag": "Que es?",
      "patterns": ["Que es la luperca?", "Que es?", "La Lloba Capitolina que es?"],
      "responses": ["La Lloba Capitolina es una estatua de bronze que representa una lloba alleitant dos nens, que segons la mitologia romana son Romul i Rem, els fundadors llegendaris de Roma."],
      "context": [""]
    },
    {
      "tag": "IMPORTANCIA",
      "patterns": ["Per que es important la Lloba Capitolina per a la ciutat de Roma?", "Quina importancia te?", "Per que es important"],
      "responses": ["La Lloba Capitolina es un simbol de la fundacio de Roma i del poder i la proteccio de la ciutat. S'ha convertit en una icona cultural i representa l'origen mitic de Roma."],
      "context": [""]
    },
    {
      "tag": "Que representen els nens",
      "patterns": ["Que representen Romul i Rem a l'escultura?", "Quina representacio tenen els nens?", "Que signifiquen els nens a l'obra"],
      "responses": ["Romul i Rem, els dos bessons lactants a l'estatua, son personatges clau en la llegenda de Roma. Segons la tradicio, Romul seria qui finalment fundaria la ciutat i es convertiria en el seu primer rei després de matar el seu germa Rem en una disputa."],
      "context": [""]
    },
    {
      "tag": "L'estil",
      "patterns": ["Quin es l'estil artistic de l'estatua de la Lloba Capitolina?", "Quin estil s'utilitza?", "Que estil es fa servir?"],
      "responses": ["L'estil de la lloba reflecteix la influencia etrusca, amb una aparença ferotge i geometrica. Les figures de Romul i Rem, en canvi, son d'un estil renaixentista mes realista i detallat."],
      "context": [""]
    },
    {
      "tag": "POLITICA I CULTURA",
      "patterns": ["L'estatua ha estat utilitzada amb finalitats politiqueres o culturals?", "Quins fins culturals o politics te?", "S'ha utilitzat a la politica o a la cultura?"],
      "responses": ["Si, la Lloba Capitolina ha estat utilitzada al llarg dels segles com a simbol de poder i unitat de Roma. Durant el regim feixista de Mussolini al segle XX, es va utilitzar amb finalitats propagandistiqueres per connectar la Roma moderna amb la seva grandesa imperial passada."],
      "context": [""]
    },
    {
      "tag": "MIDA",
      "patterns": ["Quina mida te la Lloba Capitolina?", "Quant medeix l'estatua?", "Quines mides te l'obra d'art?"],
      "responses": ["La Lloba Capitolina te una alçada d'aproximadament 75 cm i una llargada de 114 cm. Les figures de Romul i Rem son molt mes petites en comparacio amb la Lloba Capitolina."],
      "context": [""]
    }
  ]
}
```

7.2. CODI PER A ENTRENAR I CREAR LA INTEL·LIGÈNCIA ARTIFICIAL DEL XATBOT

```
import nltk
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
import json
import pickle
import numpy as np
from keras.models import Sequential
from keras.layers import Dense, Input, Dropout
from keras.optimizers import SGD
import random
words=[]
classes = []
documents = []

ignore_words = ['?', '!']
data_file = open('intents.json').read()
intents = json.loads(data_file)

nltk.download('punkt')

for intent in intents['intents']:
    for pattern in intent['patterns']:
        # tokenize each word
        w = nltk.word_tokenize(pattern)
        words.extend(w)

# add documents in the corpus
documents.append((w,intent['tag']))
#add to our classes list
if intent['tag'] not in classes:
    classes.append(intent['tag'])

nltk.download('wordnet')

# lemmatize, lower each word and remove duplicates
words = [lemmatizer.lemmatize(w.lower())
for w in words if w not in ignore_words]
words = sorted(list(set(words)))

# sort classes
classes = sorted(list(set(classes)))

# documents = combination between patterns and intents
print(len(documents),"documents")

# classes = intents
print(len(classes),"classes",classes)

# words = all words, vocabulary
print(len(words),"unique lemmatized words",words)

pickle.dump(words,open('words.pkl','wb'))
pickle.dump(classes,open('classes.pkl','wb'))

# create our training data
training = []

# create an empty array for our output
output_empty = [0] * len(classes)
# training set, bag of words for each sentence
for doc in documents:
    # initialize our bag of words
    bag = []
    # list of tokenized words for the pattern
    pattern_words = doc[0]
    # lemmatize each word - create base word, in attempt to represent related words
    pattern_words = [lemmatizer.lemmatize(word.lower()) for word in pattern_words]
    # create our bag of words array with 1, if word match found in current pattern
    for w in words:
        bag.append(1) if w in pattern_words else bag.append(0)
    # output is a '0' for each tag and '1' for current tag (for each pattern)
    output_row = list(output_empty)
    output_row[classes.index(doc[1])] = 1
    training.append([bag,output_row])

# shuffle our features and turn into np.array
random.shuffle(training)
training = np.array(training, dtype='object')

# create train and test lists. X - patterns, Y - intents
train_x = list(training[:,0])
train_y = list(training[:,1])
print("Training data created")
print(train_x)
print(train_y)

model = Sequential()
model.add(Dense(128, input_shape=(len(train_x[0]),), activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(64, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(len(train_y[0]), activation='softmax'))

sgd = SGD(learning_rate=0.01, weight_decay=1e-6, momentum=0.9, nesterov=True)
model.compile(loss='categorical_crossentropy', optimizer=sgd, metrics=['accuracy'])

# fitting and saving the model
hist = model.fit(np.array(train_x), np.array(train_y), epochs=200, batch_size=5, verbose=1)
model.save('chatbot_model.h5', hist)
print(model.summary())
print("Model created")
```

7.3. CODI DE L'APLICACIÓ D'ESCRITOR I DE LA FINESTRA DEL XATBOT

```
from tkinter import *
import customtkinter
from PIL import Image
from tkinter import messagebox

customtkinter.set_appearance_mode("light")
customtkinter.set_default_color_theme("blue")

import nltk
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()

import pickle
import numpy as np
from keras.models import load_model
model = load_model('chatbot_model.h5')

import json
import random
intents = json.loads(open('intents.json').read())
words = pickle.load(open('words.pkl', 'rb'))
classes = pickle.load(open('classes.pkl', 'rb'))

# MOSTRAR FINESTRA
def show_window1():
    root.withdraw()
    window1 = customtkinter.CTkToplevel(root)
    window1.geometry("1920x1080-10-2")

def llova_capitolina():

    def clean_up_sentence(sentence):
        # tokenize the pattern - split words into array
        sentence_words = nltk.word_tokenize(sentence)
        # stem each word - create short form for word
        sentence_words = [lemmatizer.lemmatize(word.lower()) for word in sentence_words]
        return sentence_words

    # return bag of words array: 0 or 1 for each word in the bag that exists in the sentence
    def bow(sentence, words, show_details=True):
        # tokenize the pattern
        sentence_words = clean_up_sentence(sentence)
        # bag of words - matrix of N words, vocabulary matrix
        bag = [0]*len(words)
        for s in sentence_words:
            for i,w in enumerate(words):
                if w == s:
                    # assign 1 if current word is in the vocabulary position
                    bag[i] = 1
                    if show_details:
                        print ("found in bag: %s" % w)
        return(np.array(bag))

    def predict_class(sentence, model):
        # filter out predictions below a threshold
        p = bow(sentence, words, show_details=False)
        res = model.predict(np.array([p]))[0]
        ERROR_THRESHOLD = 0.25
        results = [[i,r] for i,r in enumerate(res) if r>ERROR_THRESHOLD]
        # sort by strength of probability
        results.sort(key=lambda x: x[1], reverse=True)
        return_list = []
        for r in results:
            return_list.append({"intent": classes[r[0]], "probability": str(r[1])})
        return return_list

    def getResponse(ints, intents_json):
        tag = ints[0]['intent']
        list_of_intents = intents_json['intents']
        for i in list_of_intents:
            if(i['tag']== tag):
                result = random.choice(i['responses'])
                break
        return result

    def chatbot_response(text):
        ints = predict_class(text, model)
        res = getResponse(ints, intents)
        return res
```

```

def send():
    msg = EntryBox.get("1.0", 'end-1c').strip()
    EntryBox.delete("0.0", END)
    if msg != '':
        ChatLog.config(state=NORMAL)
        ChatLog.insert(END, "You: " + msg + '\n\n')
        ChatLog.config(foreground="#442266", font=("Verdana", 12))
        res = chatbot_response(msg)
        ChatLog.insert(END, "Bot: " + res + '\n\n')
        ChatLog.config(state=DISABLED)
        ChatLog.yview(END)

base = Tk()
base.title("Hello")
base.geometry("400x500-50-100")
base.resizable(width=FALSE, height=FALSE)
#Create Chat window
ChatLog = Text(base, bd=0, bg="white", height="8", width="50", font="Arial",)
ChatLog.config(state=DISABLED)
#Bind scrollbar to Chat window
scrollbar = Scrollbar(base, command=ChatLog.yview, cursor="heart")
ChatLog['yscrollcommand'] = scrollbar.set
#Create Button to send message
SendButton = Button(base, font=("Verdana", 12, 'bold'),
                    text="Send", width="12",
                    height=5, bd=0, bg="#3B8ED0",
                    activebackground="#36719F",
                    fg='ffffff', command= send )

#Create the box to enter message
EntryBox = Text(base, bd=0, bg="white", width="29", height="5", font="Arial")
EntryBox.bind("<Return>", send)
#Place all components on the screen
scrollbar.place(x=376, y=6, height=386)
ChatLog.place(x=6, y=6, height=386, width=370)
EntryBox.place(x=128, y=401, height=90, width=265)
SendButton.place(x=6, y=401, height=90)
base.mainloop()

# TORNAR A LA PÀGINA PRINCIPAL
def close_window1():
    window1.destroy()
    window1.update()
    root.deiconify()

# MOSTRAR FINESTRA2
def show_window2():
    window1.withdraw()
    window2 = customtkinter.CTkToplevel(window1)
    window2.geometry("1920x1080-10-2")

# MOSTRAR FINESTRA3
def show_window3():
    window2.withdraw()
    window3 = customtkinter.CTkToplevel(window2)
    window3.geometry("1920x1080-10-2")

# MOSTRAR FINESTRA4
def show_window4():
    window3.withdraw()
    window4 = customtkinter.CTkToplevel(window3)
    window4.geometry("1920x1080-10-2")

def close_window4():
    window4.destroy()
    window4.update()
    root.deiconify()

# FINESTRA4
img_lll = customtkinter.CTkImage(light_image=Image.open('FONDO_PAGFINAL_5.png'),
                                dark_image=Image.open('FONDO_PAGFINAL_5.png'),
                                size=(1920, 1080))

customtkinter.CTkLabel(window4,
                        text="",
                        image=img_lll).place(x=960, y=540, anchor=CENTER)

customtkinter.CTkButton(window4,
                        text="TANCAR",
                        height=60,
                        width=130,
                        font=("Helvetica", 24),
                        text_color="#242424",
                        command=close_window4).place(x=1252, y=935, anchor=CENTER)

```

```

# FINESTRA 3
radio_var = customtkinter.StringVar(value="other")
radio_var_2 = customtkinter.StringVar(value="other")
radio_var_3 = customtkinter.StringVar(value="other")
radio_var_4 = customtkinter.StringVar(value="other")
radio_var_5 = customtkinter.StringVar(value="other")
radio_var_6 = customtkinter.StringVar(value="other")
radio_var_7 = customtkinter.StringVar(value="other")
radio_var_8 = customtkinter.StringVar(value="other")

img_8 = customtkinter.CTkImage(light_image=Image.open('FONDO_PAG4.png'),
                                dark_image=Image.open('FONDO_PAG4.png'),
                                size=(1920, 1080))

customtkinter.CTkLabel(window3, text="", image=img_8).place(x=960, y=540, anchor=CENTER)

radio_var = customtkinter.StringVar(value="other")

customtkinter.CTkLabel(window3, text="Tornaries a visitar el", font=("Helvetica", 18)).place(x=250, y=500, anchor=CENTER)
customtkinter.CTkLabel(window3, text="museu en el futur?", font=("Helvetica", 18)).place(x=250, y=530, anchor=CENTER)
customtkinter.CTkRadioButton(window3, text="Si, segurament", variable=radio_var).place(x=170, y=570, anchor=W)
customtkinter.CTkRadioButton(window3, text="Potser", variable=radio_var).place(x=170, y=610, anchor=W)
customtkinter.CTkRadioButton(window3, text="No estic segur", variable=radio_var).place(x=170, y=650, anchor=W)
customtkinter.CTkRadioButton(window3, text="No", variable=radio_var).place(x=170, y=690, anchor=W)

customtkinter.CTkLabel(window3, text="Es aquesta la teva primera", font=("Helvetica", 18)).place(x=500, y=390, anchor=CENTER)
customtkinter.CTkLabel(window3, text="visita el museu?", font=("Helvetica", 18)).place(x=460, y=420, anchor=CENTER)

radio_var = customtkinter.StringVar(value="other")
customtkinter.CTkRadioButton(window3, text="Si", value="Si",
                              variable=radio_var_2).place(x=450, y=460, anchor=CENTER)
customtkinter.CTkRadioButton(window3, text="No", value="No",
                              variable=radio_var_2).place(x=450, y=500, anchor=CENTER)

customtkinter.CTkLabel(window3, text="Valora la teva experiència", font=("Helvetica", 18)).place(x=500, y=540, anchor=CENTER)
customtkinter.CTkLabel(window3, text="de 1 al 100", font=("Helvetica", 18)).place(x=448, y=570, anchor=CENTER)

def sliding(value):
    my_label.configure(text=str(int(value)))

my_slider = customtkinter.CTkSlider(window3,
                                     from_=0,
                                     to=100,
                                     command=sliding,
                                     orientation="horizontal",
                                     number_of_steps=10,
                                     width=250,
                                     state="normal")

my_slider.place(x=500, y=590, anchor=CENTER)
my_slider.set(0)
my_label = customtkinter.CTkLabel(window3, text="0", font=("Helvetica", 18))
my_label.place(x=500, y=620, anchor="center")

def delete():
    my_text.delete(0.0, "end")

def enviar():
    messagebox.showinfo("Desat", "La valoració ha sigut desada amb èxit.")

customtkinter.CTkLabel(window3, text="D'on ets?", font=("Helvetica", 18)).place(x=500, y=690, anchor=CENTER)

my_text = customtkinter.CTkTextbox(window3,
                                     height=50,
                                     width=220,
                                     corner_radius=10)

my_text.place(x=500, y=730, anchor=CENTER)

delete_button = customtkinter.CTkButton(master=window3,
                                         text="ESBORRAR",
                                         command=delete)

delete_button.place(x=350, y=770)

send_button = customtkinter.CTkButton(master=window3,
                                       text="DESAR",
                                       command=enviar)

send_button.place(x=500, y=770)

customtkinter.CTkLabel(window3, text="Com has conegut el museu?", font=("Helvetica", 18)).place(x=700, y=290)

radio_var = customtkinter.StringVar(value="other")

customtkinter.CTkRadioButton(window3, text="Publicitat", value="Publicitat",
                              variable=radio_var_3).place(x=700, y=330)
customtkinter.CTkRadioButton(window3, text="Xarxes socials", value="Xarxes socials",
                              variable=radio_var_3).place(x=700, y=370)
customtkinter.CTkRadioButton(window3, text="Recomanació a familiars o amics", value="Recomanació",
                              variable=radio_var_3).place(x=700, y=410)
customtkinter.CTkRadioButton(window3, text="Guies turístiques o fullets", value="Guies",
                              variable=radio_var_3).place(x=700, y=450)
customtkinter.CTkRadioButton(window3, text="Mitjans de comunicació", value="Mitjans",
                              variable=radio_var_3).place(x=700, y=490)
customtkinter.CTkRadioButton(window3, text="Oficina de turisme", value="Oficina",
                              variable=radio_var_3).place(x=700, y=530)
customtkinter.CTkRadioButton(window3, text="Altres", value="Altres",
                              variable=radio_var_3).place(x=700, y=570)

customtkinter.CTkLabel(window3, text="En quin grup d'edat et trobes?", font=("Helvetica", 18)).place(x=700, y=620)

customtkinter.CTkRadioButton(window3, text="Menys de 18", value="més 18",
                              variable=radio_var_4).place(x=700, y=660)
customtkinter.CTkRadioButton(window3, text="Entre 18-25", value="18-25",
                              variable=radio_var_4).place(x=700, y=700)
customtkinter.CTkRadioButton(window3, text="Entre 26-35", value="26-35",
                              variable=radio_var_4).place(x=700, y=740)
customtkinter.CTkRadioButton(window3, text="Entre 36-45", value="36-45",
                              variable=radio_var_4).place(x=700, y=780)
customtkinter.CTkRadioButton(window3, text="Entre 46-55", value="46-55",
                              variable=radio_var_4).place(x=700, y=820)
customtkinter.CTkRadioButton(window3, text="Més de 55", value="més 55",
                              variable=radio_var_4).place(x=700, y=860)

```

```

customtkinter.CTkLabel(window3, text="Recomanaries la teva visita?", font=("Helvetica", 18)).place(x=1000, y=290)

customtkinter.CTkRadioButton(window3, text="Si", variable=radio_var).place(x=1000, y=340)

customtkinter.CTkRadioButton(window3, text="No", variable=radio_var).place(x=1000, y=380)

customtkinter.CTkLabel(window3, text="A qui li recomanaries vindre?", font=("Helvetica", 18)).place(x=1000, y=430)

radio_var = customtkinter.StringVar(value="other")

customtkinter.CTkRadioButton(window3, text="Familiars", variable=radio_var_5).place(x=1000, y=470)
customtkinter.CTkRadioButton(window3, text="Amics", variable=radio_var_5).place(x=1000, y=510)
customtkinter.CTkRadioButton(window3, text="Turistes", variable=radio_var_5).place(x=1000, y=550)
customtkinter.CTkRadioButton(window3, text="Amants de la història", variable=radio_var_5).place(x=1000, y=590)
customtkinter.CTkRadioButton(window3, text="Compans de feina", variable=radio_var_5).place(x=1000, y=630)
customtkinter.CTkRadioButton(window3, text="Estudiants", variable=radio_var_5).place(x=1000, y=670)
customtkinter.CTkRadioButton(window3, text="Altres", variable=radio_var_5).place(x=1000, y=710)

customtkinter.CTkLabel(window3, text="Valoració de l'ajuda", font=("Helvetica", 18)).place(x=1250, y=290)
customtkinter.CTkLabel(window3, text="del xatbot", font=("Helvetica", 18)).place(x=1250, y=320)

def sliding_2(value):
    my_label_2.configure(text=str(int(value)))

my_slider = customtkinter.CTkSlider(window3,
                                    from_=0,
                                    to=100,
                                    command=sliding_2,
                                    orientation="horizontal",
                                    number_of_steps=10,
                                    width=250,
                                    state="normal")

my_slider.place(x=1250, y=360)
my_slider.set(0)
my_label_2 = customtkinter.CTkLabel(window3, text="0", font=("Helvetica", 18))
my_label_2.place(x=1370, y=380)

def delete_2():
    my_text_2.delete(0.0, "end")

def enviar_2():
    messagebox.showinfo("Desat", "La valoració ha sigut desada amb èxit.")

customtkinter.CTkLabel(window3, text="Siusplau, detalla els motius", font=("Helvetica", 18)).place(x=1250, y=440)
customtkinter.CTkLabel(window3, text="de la teva valoració", font=("Helvetica", 18)).place(x=1250, y=470)

my_text_2 = customtkinter.CTkTextbox(window3,
                                     height=150,
                                     width=220,
                                     corner_radius=1)

my_text_2.place(x=1250, y=500)

delete_button = customtkinter.CTkButton(master=window3,
                                       text="ESBORRAR",
                                       command=delete_2)

delete_button.place(x=1200, y=690)

send_button = customtkinter.CTkButton(master=window3,
                                       text="DESAR",
                                       command=enviar_2)

send_button.place(x=1350, y=690)

customtkinter.CTkLabel(window3, text="Què t'ha semblat l'exposició", font=("Helvetica", 18)).place(x=1550, y=290)
customtkinter.CTkLabel(window3, text="de la Lloba Capitolina?", font=("Helvetica", 18)).place(x=1550, y=320)
customtkinter.CTkRadioButton(window3, text="Molt interessant i enriquidora", variable=radio_var_6).place(x=1550, y=370)
customtkinter.CTkRadioButton(window3, text="Interessant", variable=radio_var_6).place(x=1550, y=410)
customtkinter.CTkRadioButton(window3, text="Regular, m'esperava més", variable=radio_var_6).place(x=1550, y=450)
customtkinter.CTkRadioButton(window3, text="Poc interessant", variable=radio_var_6).place(x=1550, y=490)
customtkinter.CTkRadioButton(window3, text="No m'ha agradat gens", variable=radio_var_6).place(x=1550, y=530)

customtkinter.CTkLabel(window3, text="T'han resultat útils i comprensibles les", font=("Helvetica", 18)).place(x=1550, y=600)
customtkinter.CTkLabel(window3, text="explicacions de les obres exposades?", font=("Helvetica", 18)).place(x=1550, y=640)
customtkinter.CTkRadioButton(window3, text="Molt útils i fàcils de comprendre", variable=radio_var_7).place(x=1550, y=680)
customtkinter.CTkRadioButton(window3, text="Útils i comprensibles", variable=radio_var_7).place(x=1550, y=720)
customtkinter.CTkRadioButton(window3, text="Regulars, amb algunes parts confuses", variable=radio_var_7).place(x=1550, y=760)
customtkinter.CTkRadioButton(window3, text="Poc útils o difícils de comprendre", variable=radio_var_7).place(x=1550, y=800)
customtkinter.CTkRadioButton(window3, text="No em van resultar útils ni comprensibles", variable=radio_var_7).place(x=1550, y=840)

img_10 = customtkinter.CTkImage(light_image=Image.open('fletxa_dreta.png'),
                                dark_image=Image.open('fletxa_dreta.png'),
                                size=(50, 50))

customtkinter.CTkButton(window3,
                        image=img_10,
                        text="",
                        command=show_window4).place(x=1550, y=900)

# FINESTRA2
img_7 = customtkinter.CTkImage(light_image=Image.open("FONDO_PAG3.png"),
                                dark_image=Image.open('FONDO_PAG3.png'),
                                size=(1920, 1080))

customtkinter.CTkLabel(window2, text="", image=img_7).place(x=960, y=540, anchor=CENTER)

customtkinter.CTkButton(window2,
                        text="Continuar amb la valoració",
                        command=show_window3,
                        height=80,
                        width=150,
                        font=("Helvetica", 24),
                        corner_radius=14,
                        text_color="#242424").place(x=960, y=900, anchor=CENTER)

```

```

# FINESTRAL
img_7 = customtkinter.CTkImage(light_image=Image.open('pagina2.png'),
                                dark_image=Image.open('pagina2.png'),
                                size=(1920,1080))
label = customtkinter.CTkLabel(window1,text="",image=img_7)
label.place(x=960,y=540,anchor=CENTER)

customtkinter.CTkButton(master=window1,
                        text="LOBA CAPITOLINA",
                        command=llova_capitolina,
                        height=50,
                        width=100,
                        font=("Helvetica",24),
                        corner_radius=14,
                        text_color="#242424").place(x=420,y=520, anchor=CENTER)

customtkinter.CTkButton(master=window1,
                        text="SARCÓFAG DEL PEDAGOG",
                        height=50,
                        width=100,
                        font=("Helvetica",24),
                        corner_radius=14,
                        text_color="#242424").place(x=980,y=520, anchor=CENTER)

customtkinter.CTkButton(master=window1,
                        text="MOSAIC DE LA MEDUSA",
                        height=50,
                        width=100,
                        font=("Helvetica",24),
                        corner_radius=14,
                        text_color="#242424").place(x=1520,y=520, anchor=CENTER)

customtkinter.CTkButton(master=window1,
                        text="LAUDA SEPULCRAL D'ÒPTIM",
                        height=50,
                        width=100,
                        font=("Helvetica",24),
                        corner_radius=14,
                        text_color="#242424").place(x=420,y=960, anchor=CENTER)

customtkinter.CTkButton(master=window1,
                        text="LUCI VER",
                        height=50,
                        width=100,
                        font=("Helvetica",24),
                        corner_radius=14,
                        text_color="#242424").place(x=980,y=960, anchor=CENTER)

customtkinter.CTkButton(master=window1,
                        text="CLÍPEUS DE JÚPITER",
                        height=50,
                        width=100,
                        font=("Helvetica",24),
                        corner_radius=14,
                        text_color="#242424").place(x=1520,y=960, anchor=CENTER)

img_1 = customtkinter.CTkImage(light_image=Image.open('fletxa_esquerra.png'),
                                dark_image=Image.open('fletxa_esquerra.png'),
                                size=(50,50))

customtkinter.CTkButton(window1,
                        image=img_1,
                        text="",
                        command=close_window1).place(x=100, y=150, anchor=CENTER)

img_2 = customtkinter.CTkImage(light_image=Image.open('fletxa_dreta.png'),
                                dark_image=Image.open('fletxa_dreta.png'),
                                size=(50,50))

customtkinter.CTkButton(window1,
                        image=img_2,
                        text="",
                        command=show_window2).place(x=1800, y=150, anchor=CENTER)

# FINESTRA PRINCIPAL
root = customtkinter.CTk()
root.geometry("1920x1080-10-2")

img_4 = customtkinter.CTkImage(light_image=Image.open('FONDO2.png'),
                                dark_image=Image.open('FONDO2.png'),
                                size=(1920,1080))

label = customtkinter.CTkLabel(root,text="",image=img_4)
label.place(x=960,y=540,anchor=CENTER)

customtkinter.CTkButton(master=root,
                        text="Continuar amb la visita",
                        command=show_window1,
                        height=80,
                        width=150,
                        font=("Helvetica",24),
                        corner_radius=14,
                        text_color="#242424").place(x=320,y=900, anchor=CENTER)

root.mainloop()

```