

**Denis Shatalin Vetojin**

**SISTEMA ENCASTAT PER A LA TRADUCCIÓ DE L'ALFABET  
DACTILOLÒGIC**

**TREBALL DE FI DE GRAU**

**dirigit per David Gámez Alari**

**Grau d'Enginyeria Informàtica**



**UNIVERSITAT ROVIRA I VIRGILI**

**Tarragona**

**2025**

**Resum.**

Aquest projecte consisteix en el desenvolupament d'un guant intel·ligent capaç de traduir el llenguatge de signes a caràcters alfabètics mitjançant l'ús de sensors i un microcontrolador. La motivació principal d'aquest treball és contribuir a la millora de la comunicació entre persones amb discapacitat auditiva i aquelles que no coneixen el llenguatge de signes, apropant així la tecnologia a la inclusió social.

El sistema es basa en la col·locació de sensors de flexibilitat als dits del guant i un sensor d'orientació per detectar la posició i la inclinació de la mà. A través d'un microcontrolador, les dades recollides pels sensors són processades i comparades amb una base de dades interna per identificar la lletra corresponent.

La metodologia ha inclòs el disseny electrònic, la programació del microcontrolador, la definició d'un diccionari de gestos, i la realització de proves amb diversos usuaris per calibrar i validar la precisió del sistema.

Els resultats obtinguts mostren una capacitat del sistema per reconèixer lletres individuals amb una precisió acceptable en condicions de laboratori, amb marge de millora mitjançant tècniques de machine learning o bases de dades més àmplies. El projecte demostra la viabilitat d'aquest tipus de solucions com a eina educativa, comunicativa i inclusiva.

**Resumen.**

Este proyecto se basa en el desarrollo de un guante inteligente capaz de traducir el lenguaje de signos a caracteres alfabéticos utilizando sensores y una placa microcontroladora. La principal motivación del trabajo es contribuir a la mejora de la comunicación entre personas con discapacidad auditiva y aquellas que no conocen el lenguaje de signos, acercando la tecnología a la inclusión social.

El sistema utiliza sensores de flexión colocados en los dedos del guante y un sensor de orientación para detectar la posición e inclinación de la mano. Con un microcontrolador, los datos se procesan y se comparan con una base de datos interna para identificar la letra correspondiente.

La metodología ha consistido en el diseño electrónico, la programación de la placa, la definición de un diccionario de gestos y pruebas de validación con usuarios.

Los resultados obtenidos indican que el sistema puede reconocer letras individuales con un nivel aceptable de precisión en un entorno controlado, abriendo la puerta a futuras mejoras y aplicaciones tanto educativas como comunicativas.

**Abstract.**

This project involves the development of a smart glove capable of translating sign language into alphabetic characters using sensors and a microcontroller. The main objective is to improve communication between hearing-impaired individuals and those unfamiliar with sign language, promoting social inclusion through technology.

The system integrates flex sensors on the fingers and an orientation sensor to detect hand movements. Data is processed by the microcontroller and matched against a gesture dictionary to identify the corresponding letter.

Initial results show acceptable accuracy in detecting individual letters in a controlled environment. The project proves the viability of this solution for educational and communicative purposes. This project involves the development of a smart glove capable of translating sign language into alphabetic characters using sensors and a microcontroller. The main objective is to improve communication between hearing-impaired individuals and those unfamiliar with sign language, promoting social inclusion through technology.

The system integrates flex sensors on the fingers and an orientation sensor to detect hand movements. Data is processed and matched against a gesture dictionary to identify the corresponding letter.

Initial results show acceptable accuracy in detecting individual letters in a controlled environment. The project proves the viability of this solution for educational and communicative purposes.

# Índex

<b>1</b>	<b>INTRODUCCIÓ</b>	<b>8</b>
<b>2</b>	<b>DESCRIPCIÓ GENERAL DEL PROJECTE</b>	<b>9</b>
2.1	NECESSITATS IDENTIFICADES	11
2.2	PREVISIONS D'ÚS	11
<b>3</b>	<b>REQUISITS</b>	<b>12</b>
3.1	COMPONENTS NECESSARIS	12
3.2	REQUISITS FUNCIONALS	13
3.3	REQUISITS NO FUNCIONALS	15
<b>4</b>	<b>ANÀLISI DELS REQUISITS FUNCIONALS</b>	<b>17</b>
4.1	DIAGRAMA CLASSES GENERAL	17
4.2	ANÀLISI PER CAS D'ÚS	18
4.2.1	<i>Calibratge del sensor d'orientació</i>	18
4.2.2	<i>Calibratge dels sensors de flexibilitat</i>	19
4.2.3	<i>Captura de dades amb sensors</i>	20
4.2.4	<i>Predicció de la lletra amb K-NN</i>	21
4.2.5	<i>Visualització de la lletra</i>	22
4.2.6	<i>Entrenament del diccionari de gestos</i>	23
<b>5</b>	<b>DISSENY</b>	<b>24</b>
5.1	ARQUITECTURA GENERAL DEL SISTEMA	24
5.2	ELECCIÓ DELS COMPONENTS	24
5.3	DISSENY DE LA INTERFÍCIE AMB L'USUARI	26
5.4	ESQUEMA DE CONNEXIONS ELECTRÒNIQUES	27
5.5	DISSENY DE LA PERSISTÈNCIA DE DADES	28
5.6	ESTRATÈGIA DE COMPARACIÓ DE GESTOS	29
5.7	CONCLUSIÓ	29
<b>6</b>	<b>IMPLEMENTACIÓ</b>	<b>30</b>
6.1	ENTORN DE DESENVOLUPAMENT	30
6.1.1	<i>Configuració de PlatformIO i llibreries utilitzades</i>	31
6.2	SPARKFUN THING PLUS ESP32 WROOM-32E	31
6.2.1	<i>Descripció funcional</i>	33
6.2.2	<i>Integració al projecte</i>	33
6.3	SENSORS FLEX	33
6.3.1	<i>Descripció funcional</i>	34
6.3.2	<i>Integració al projecte</i>	34
6.4	SENSOR D'ORIENTACIÓ MPU-6050	35
6.4.1	<i>Descripció funcional</i>	35
6.4.2	<i>Lectura de dades i conversió</i>	35
6.4.3	<i>Ús de <code>getMotion6()</code> i <code>int16_t</code></i>	36
<b>7</b>	<b>PROCÉS ITERATIU DE DESENVOLUPAMENT I MILLORA</b>	<b>37</b>
7.1	LECTURA INICIAL DELS SENSORS DE FLEXIBILITAT	37
7.2	FIXACIÓ DELS SENSORS I PROVES INICIALS	37
7.3	IMPLEMENTACIÓ D'UN SISTEMA DE CALIBRATGE I MILLORA EN FIXACIÓ	38
7.4	INTEGRACIÓ DEL SENSOR D'ORIENTACIÓ	38
7.5	DETECCIÓ DE LLETRES AMB MOVIMENT I EL CAS ESPECIAL DE LA Y	38
7.6	IMPLEMENTACIÓ D'UN SISTEMA K-NN PER MILLORAR LA DETECCIÓ	39
7.7	FUNCIONAMENT DEL SISTEMA K-NN	39
7.8	MILLORA DE LA FIABILITAT I RESULTATS OBTINGUTS	39
7.9	APLICACIÓ MÒBIL PER A DISPOSITIUS ANDROID	40
7.9.1	<i>Funcionalitats principals</i>	40

7.9.2	<i>Desenvolupament tècnic</i> .....	41
7.10	MILLORES POSSIBLES .....	41
<b>8</b>	<b>REFLEXIÓ ÈTICA I RESPONSABILITAT SOCIAL .....</b>	<b>43</b>
8.1	IGUALTAT .....	43
8.2	MEDI AMBIENT .....	43
8.3	RESPONSABILITAT SOCIAL .....	43
8.4	ÈTICA.....	44
<b>9</b>	<b>CONCLUSIONS .....</b>	<b>45</b>
<b>10</b>	<b>REFERÈNCIES.....</b>	<b>46</b>
<b>11</b>	<b>ANNEX I. POSADA EN MARXA I GUIA D'ÚS.....</b>	<b>47</b>
11.1	CONFIGURACIÓ DE PLATFORMIÓ I LLIBRERIES .....	47
11.2	COMPILACIÓ I CÀRREGA DEL FIRMWARE A LA PLACA ESP32 .....	47
11.3	EXECUCIÓ DE L'APLICACIÓ MÒBIL.....	48
11.4	FUNCIONAMENT DE L'APLICACIÓ I CALIBRATGE .....	48
11.5	REINICI DEL SISTEMA .....	48

## Índex de taules

TAULA 1. NECESSITATS IDENTIFICADES.....	11
TAULA 2. REQUISITS FUNCIONALS .....	15
TAULA 3. REQUISITS NO FUNCIONALS .....	16

## Índex de figures

FIGURA 1. ESQUEMA BÀSIC DE FUNCIONAMENT DEL SISTEMA .....	10
FIGURA 2. ALFABET DACTILOLÒGIC DE LA LLENGUA DE SIGNES CATALANA (LSC) [1] .....	10
FIGURA 3. DIAGRAMA DE CLASSES GENERAL .....	17
FIGURA 4. DIAGRAMA DE CLASSES CAS D'ÚS 01 .....	18
FIGURA 5. DIAGRAMA DE SEQÜÈNCIES CAS D'ÚS 01 .....	18
FIGURA 6. DIAGRAMA DE CLASSES CAS D'ÚS 03 .....	20
FIGURA 7. DIAGRAMA DE SEQÜÈNCIES CAS D'ÚS 03 .....	20
FIGURA 8. DIAGRAMA DE CLASSES CAS D'ÚS 04 .....	21
FIGURA 9. DIAGRAMA DE SEQÜÈNCIES CAS D'ÚS 04 .....	21
FIGURA 10. DIAGRAMA DE CLASSES CAS D'ÚS 05 .....	22
FIGURA 11. DIAGRAMA DE SEQÜÈNCIES CAS D'ÚS 05 .....	22
FIGURA 12. DIAGRAMA DE CLASSES CAS D'ÚS 06 .....	23
FIGURA 13. DIAGRAMA DE SEQÜÈNCIES CAS D'ÚS 06 .....	23
FIGURA 14. ESQUEMA DETALLAT DE CONNEXIONS ENTRE ELS SENSORS, LA PLACA ESP32 I LA PROTOBOARD.....	27
FIGURA 15. ESQUEMA DE CONNEXIONS DELS SENSORS AMB UN MICROCONTROLADOR (EXEMPLE FUNCIONAL) .....	28
FIGURA 16. IMATGE D'UNA SPARKFUN THING PLUS ESP32 WROOM-32E [2] .....	31
FIGURA 17. ESQUEMA DE PINS DEL ESP32 [3] .....	32
FIGURA 18. ESQUEMA DEL ESP32 [4] .....	32
FIGURA 19. SENSOR DE FLEXIBILITAT [5] .....	34
FIGURA 20. SENSOR D'ORIENTACIÓ MPU-6050 [7] .....	35
FIGURA 21. CONFIGURACIÓ BÀSICA DEL FITXER PLATFORMIO.INI.....	47



## 1 Introducció

La comunicació és una de les capacitats més importants de l'ésser humà. Ens permet expressar idees, emocions i informació, i resulta essencial per participar activament en la societat. Quan hi ha barreres comunicatives, apareixen situacions de desconnexió que poden afectar greument la inclusió i la igualtat d'oportunitats. És el cas de moltes persones amb discapacitat auditiva, que sovint es troben amb dificultats per fer-se entendre, especialment quan els seus interlocutors no coneixen la llengua de signes.

La llengua de signes és el mitjà de comunicació habitual per a una gran part de la comunitat sorda. Tot i això, el seu ús no està generalitzat i la majoria de la població no la domina. Davant d'aquesta realitat, la tecnologia pot tenir un paper clau. L'aparició de dispositius portables, microcontroladors de baix cost i sensors assequibles fa possible pensar en solucions accessibles que poden millorar la qualitat de vida de moltes persones.

Aquest treball final de grau té com a objectiu el disseny i desenvolupament d'un sistema que permeti traduir gestos de la mà en lletres de l'alfabet, mitjançant un guant intel·ligent amb sensors integrats i una aplicació mòbil. La intenció és crear una eina funcional, econòmica i fàcil d'utilitzar, que pugui ajudar en la comunicació diària, però també en entorns formatius o d'aprenentatge.

En resum, aquest TFG vol demostrar que amb eines accessibles i una bona planificació, és possible desenvolupar solucions útils i pràctiques. Més enllà de la part tècnica, la voluntat del projecte és clara: fer que la tecnologia serveixi per apropar persones i facilitar la comunicació en entorns on, sovint, hi ha barreres difícils de superar.

## 2 Descripció general del projecte

La comunicació és una de les funcions més essencials per al desenvolupament personal, social i emocional dels éssers humans. Ens permet compartir pensaments, emocions, instruccions i coneixements, i constitueix la base de la interacció humana. En aquest context, quan sorgeixen barreres comunicatives, ja sigui per qüestions lingüístiques, culturals o sensorials, es redueix la capacitat de participació plena i equitativa de les persones afectades.

Una de les situacions més habituals en què aquesta barrera es fa evident és entre persones amb discapacitat auditiva i persones oients que no dominen la llengua de signes. Malgrat els avenços socials i l'augment de la sensibilització, la llengua de signes encara no és coneguda per la majoria de la població, i això limita l'autonomia i les oportunitats de comunicació de moltes persones sordes. Aquesta limitació obstaculitza àmbits com l'educació, el treball, l'atenció sanitària o la vida quotidiana.

Amb l'objectiu de reduir aquesta bretxa comunicativa, aquest treball de final de grau proposa el desenvolupament d'una solució tecnològica accessible, senzilla i portable que actuï com a intermediari entre les persones que utilitzen la llengua de signes i aquelles que no la coneixen. La proposta consisteix en un sistema capaç de reconèixer gestos de la mà associats a l'alfabet manual i traduir-los automàticament en lletres visibles en una aplicació mòbil, afavorint així la transmissió de missatges de manera més fluida i universal.

El projecte s'adreça especialment a situacions de comunicació directa, com ara una conversa entre una persona sorda i una persona oient en un espai públic, una consulta mèdica, una situació educativa o un entorn domèstic. També pot tenir un ús formatiu, com a eina de suport per a l'aprenentatge de l'alfabet manual o per a la pràctica de la llengua de signes.

La idea és construir un sistema físic i digital combinat que, a partir dels moviments de la mà de l'usuari, sigui capaç d'identificar en temps real la lletra corresponent. Aquest sistema es basa en tres elements fonamentals:

- Un guant funcional que permet detectar els moviments i les posicions dels dits de la mà.
- Un dispositiu electrònic central que interpreta les dades obtingudes del guant i les tradueix a caràcters textuais.
- Una aplicació mòbil que mostra de forma clara i accessible la informació reconeguda, permetent a l'usuari veure les lletres, paraules o frases resultants.

Alfabet bàsic					
a	b	c	d	e	f
g	h	i	j	k	l
m	n	o	p	q	r
s	t	u	v	w	x
y	z				

Figura 2. Alfabet dactilològic de la llengua de signes catalana (LSC) [1]

El funcionament del sistema es pot resumir en aquest esquema senzill:

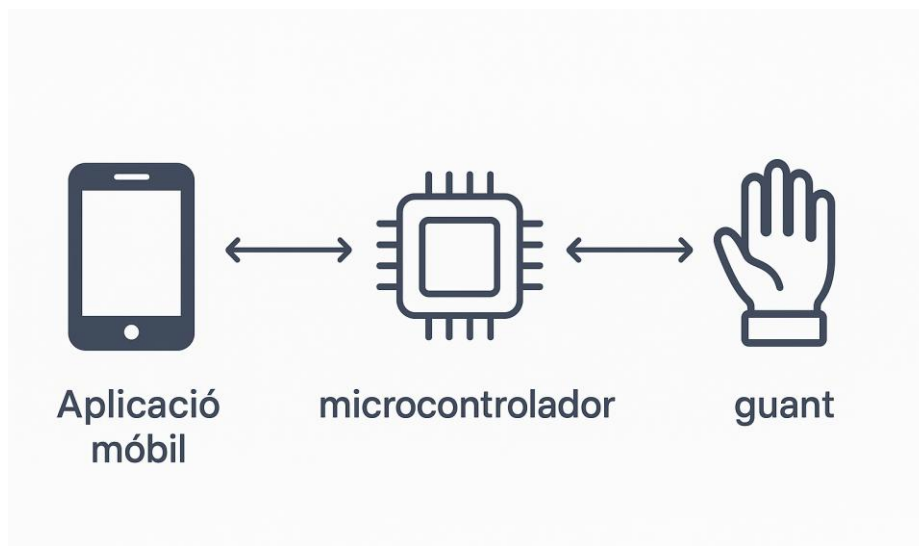


Figura 1. Esquema bàsic de funcionament del sistema

Amb aquest enfocament, el projecte busca combinar la utilitat pràctica amb la viabilitat tècnica i econòmica, aprofitant la disponibilitat actual de materials electrònics de baix cost i la potència de les plataformes mòbils. Així, es planteja com una solució

assequible, adaptable i oberta a millores futures, amb potencial per ser utilitzada en àmbits diversos i contribuir a la inclusió social.

Aquest document descriu detalladament el procés de disseny i desenvolupament d'aquest sistema, incloent-hi la seva arquitectura, les decisions de disseny i la implementació tècnica.

## 2.1 Necessitats identificades

Les principals necessitats detectades que han donat origen al projecte són:

- Necessitat d'accessibilitat comunicativa: Absència de sistemes assequibles que tradueixin llenguatge de signes en temps real.
- Portabilitat i autonomia: Dispositius que no depenguin de mòbils, aplicacions o connexió a internet.
- Fàcil adaptació: Solució que es pugui adaptar a qualsevol mà i calibrar de forma senzilla.
- Cost baix: Fer servir components econòmics per garantir que el projecte sigui reproducible i escalable.

Necessitat detectada	Objectiu de disseny corresponent
Dificultat per a persones oients de comprendre el llenguatge de signes	Crear un sistema que tradueixi gestos a lletres de forma automàtica i en temps real
Solució portable i autònoma, sense dependència de connexió constant	Desenvolupar un dispositiu que funcioni de manera independent i sense connexió a internet
Dispositiu accessible a nivell econòmic	Utilitzar components de baix cost
Necessitat d'adaptabilitat a diferents usuaris i mans	Implementar un sistema de calibratge personalitzat per a cada usuari
Interfície d'ús senzilla per a persones no tècniques	Crear una aplicació mòbil amb funcions clares i intuïtives

**Taula 1.** Necessitats identificades

## 2.2 Previsions d'ús

El guant traductor està pensat per reconèixer lletres de l'alfabet en llengua de signes i mostrar-les en una interfície visual. S'espera que pugui ser utilitzat per:

- Estudiants amb discapacitat auditiva per facilitar la seva comunicació a l'aula.
- Professionals no signants (mestres, infermers, educadors socials...) que vulguin comprendre missatges bàsics.
- Famílies que volen interactuar amb un membre amb dificultats d'audició o parla.
- Persones interessades a aprendre llengua de signes, com a eina educativa.
- El sistema no pretén substituir la figura de l'interpret, però sí oferir una eina de suport per a comunicacions senzilles i entorns informals.

### 3 Requisites

En aquesta secció es documenten els requisits que ha de complir el sistema desenvolupat, tant des del punt de vista funcional com no funcional. Els requisits funcionals fan referència a les funcionalitats que ha de proporcionar el guant traductor, mentre que els requisits no funcionals estableixen limitacions i característiques de qualitat del sistema.

Tots aquests requisits han estat definits tenint en compte l'objectiu principal del projecte: facilitar la comunicació entre persones amb discapacitat auditiva i persones que no dominen el llenguatge de signes, mitjançant una solució tècnica senzilla, fiable i portable.

#### 3.1 Components necessaris

Per a dur a terme el projecte de traducció de gestos manuals en lletres, ha estat necessari comptar amb una sèrie de components físics i digitals que permeten la captació, processament i visualització de les dades en temps real.

Els principals components físics i elements digitals requerits són:

- Guant adaptat amb sensors:  
És l'element clau per captar el moviment dels fits i la posició de la mà. Per tal que sigui funcional, s'hi integren sensors específics per detectar:
  - Flexió de dits utilitzant sensors que canvien de resistència en funció de la curvatura.
  - Orientació i moviment angular, mitjançant un sensor capaç de captar inclinacions i girs de la mà en diferents eixos.
  
- Mòdul de processament integrat  
Cal un dispositiu capaç de:
  - Llegir dades provinents dels sensors.
  - Processar aquestes dades en temps real.
  - Aplicar un algorisme de reconeixement de patrons.
  - Tindre Bluetooth o Wi-Fi integrat per enviar els resultats a l'aplicació mòbil.
  
- Aplicació mòbil per Android  
És la interfície final de l'usuari, i ha de ser capaç de:
  - Connectar-se al mòdul de processament mitjançant Bluetooth.
  - Mostrar en temps real les lletres reconegudes.
  - Permetre calibratges i gestionar suggeriments de paraules.
  
- Algorisme de classificació  
Per tal de reconèixer les lletres, el sistema utilitzarà un algorisme senzill de classificació, que compara els gestos nous amb un conjunt de gestos prèviament enregistrats per determinar el gest que s'està realitzant.

### 3.2 Requisits funcionals

Els requisits funcionals s'han extret a partir de l'anàlisi de les necessitats de l'usuari i les funcionalitats esperades del sistema. Aquests requisits han estat especificats mitjançant casos d'ús que permeten representar de forma estructurada les accions que realitzen els usuaris i el sistema.

#### Cas d'ús 1: Calibratge del sensor d'orientació

- Actor principal: Usuari.
- Actor secundari: Aplicació mòbil, Sistema.
- Precondició: El sistema està encès i inicialitzat correctament.
- Postcondició: S'han guardat la posició inicial del sensor d'orientació.

Descripció:

1. L'usuari inicia el calibratge dels sensors de flexibilitat des de l'aplicació mòbil.
2. L'aplicació envia una comanda al sistema per iniciar el calibratge.
3. El sistema demana a l'usuari que mantingui la mà en repòs per calibrar la posició inicial del sensor d'orientació.
4. L'usuari prem el botó OK.
5. El sistema registra els valors d'inclinació (roll, pitch i yaw) del sensor d'orientació.
6. Es guarden les dades calibrades per a ús posterior.

#### Cas d'ús 2: Calibratge dels sensors de flexibilitat

- Actor principal: Usuari.
- Actor secundari: Aplicació mòbil, Sistema.
- Precondició: El sistema està encès i inicialitzat correctament.
- Postcondició: S'han guardat els valors mínim i màxim de cada sensor de flexibilitat.

Descripció:

7. L'usuari inicia el calibratge dels sensors de flexibilitat des de l'aplicació mòbil.
8. L'aplicació envia una comanda al sistema per iniciar el calibratge.
9. El sistema demana a l'usuari que mantingui la mà oberta i després que flexioni cada dit.
10. L'usuari prem el botó OK per cada pas del calibratge.
11. El sistema registra per a cada sensor els valors màxims i mínims.
12. Es guarden les dades calibrades per a ús posterior.

#### Cas d'ús 3: Captura de dades amb sensors

- Actor principal: Sistema.
- Actor secundari: Sensors de flexibilitat, Sensor d'orientació.
- Precondició: El sistema ha d'estar encès i en mode de reconeixement.
- Postcondició: S'ha generat una nova estructura de dades (Lectura) amb totes les dades capturades.

Descripció:

1. El sistema accedeix als sensors de flexibilitat i llegeix els valors analògics dels cinc dits.
2. El sistema accedeix al sensor d'orientació per obtenir els valors d'orientació (pitch, roll, yaw) i velocitat de gir.
3. El sistema crea un objecte (Lectura) amb les dades obtingudes.
4. El sistema passa la lectura al classificador K-nn o desa temporalment per a comparacions posteriors.

#### **Cas d'ús 4: Predicció de la lletra amb Classificador K-NN**

- Actor principal: Sistema.
- Actor secundari: Classificador KNN, Lectura.
- Precondició: S'ha capturat una lectura vàlida dels sensors.
- Postcondició: El sistema ha retornat la lletra més probable.

Descripció:

1. El sistema rep una estructura Lectura amb dades de flexió, orientació i gir.
2. El sistema passa l'estructura al Classificador KNN.
3. El classificador KNN calcula la distància euclidiana entre la lectura i cada mostra entrenada.
4. El classificador KNN selecciona les K mostres més properes.
5. El classificador KNN fa una votació per determinar la lletra majoritària.
6. El classificador KNN retorna la lletra predita al sistema.

#### **Cas d'ús 5: Visualització de la lletra**

- Actor principal: Sistema.
- Actor secundari: Aplicació mòbil, Usuari.
- Precondició: El sistema ha predit una lletra vàlida.
- Postcondició: La lletra s'ha mostrat visualment a l'usuari.

Descripció:

1. El sistema crida el bloc de codi necessari per mostrar la lletra predita.
2. El sistema transmet la lletra que es vol mostrar per Bluetooth a l'aplicació mòbil.
3. L'aplicació mòbil rep la lletra i actualitza la interfície gràfica.
4. L'usuari veu la lletra reconeguda en temps real.

#### **Cas d'ús 6: Entrenament del diccionari de gestos**

- Actor principal: Desenvolupador.
- Actor secundari: Sistema, Classificador KNN.
- Precondició: El sistema ha d'estar connectat per port sèrie a un ordinador i en mode entrenament.
- Postcondició: S'ha afegit una nova mostra al conjunt d'entrenament del classificador KNN, guardada a la memòria del sistema.

Descripció:

1. El desenvolupador inicia el mode d'entrenament des del port sèrie.
2. El sistema captura la lectura completa dels sensors (flexions, orientació i gir).

3. A partir de les dades capturades, es crea un nou objecte Gest, etiquetat amb la lletra que l'usuari hagi indicat.
4. Aquesta memòria s'afegeix a l'estructura de mostres entrenades del Classificador KNN.
5. Les dades queden disponibles fins que es faci una nova sessió d'entrenament.

<b>Codi</b>	<b>Nom del requisit</b>	<b>Descripció</b>
RF01	Captura de dades amb sensors	El sistema ha de llegir els valors dels sensors de flexibilitat i orientació.
RF02	Reconeixement de la lletra	El sistema ha de comparar les dades amb el diccionari intern per identificar la lletra.
RF03	Visualització de la lletra	La lletra detectada s'ha de mostrar per pantalla.
RF04	Calibratge del guant	El sistema ha de permetre calibrar els sensors per adaptar-se a la mà de l'usuari.
RF05	Actualització del diccionari de gestos	Ha de ser possible modificar o ampliar el conjunt de gestos reconeguts.

Taula 2. Requisits funcionals

### 3.3 Requisits no funcionals

Els requisits no funcionals defineixen propietats generals que ha de complir el sistema, com ara la rapidesa de resposta, la compatibilitat amb el maquinari emprat o la seva escalabilitat futura. Aquests requisits no determinen accions concretes, sinó condicions sota les quals han d'executar-se les funcionalitats.

<b>Codi</b>	<b>Nom del requisit</b>	<b>Descripció</b>
RNF01	Compatibilitat	El sistema ha de ser compatible amb ESP32 i sensors de 3.3V.
RNF02	Temps de resposta	La detecció d'un gest no ha de superar els 200 ms.
RNF03	Portabilitat	El dispositiu ha de ser lleuger i còmode per a ús portàtil.

RNF04	Facilitat de calibratge	El calibratge ha de poder fer-se amb un mínim de passos per part de l'usuari.
RNF05	Robustesa	El sistema ha de continuar funcionant correctament si un sensor falla temporalment.
RNF06	Escalabilitat	El codi ha d'estar preparat per afegir noves lletres o gestos en el futur.
RNF07	Ús sense connexió	El sistema ha de funcionar sense necessitat de connexió a Internet.

**Taula 3.** Requisits no funcionals

## 4 Anàlisi dels requisits funcionals

L'anàlisi dels requisits funcionals té com a objectiu descompondre i modelar les funcionalitats del sistema des d'una perspectiva estructurada, mitjançant l'ús de diagrames UML. Aquest procés ajuda a comprendre com interactuen els diferents components del sistema i com responen davant les accions dels usuaris.

En aquest projecte, s'ha realitzat l'anàlisi a partir dels casos d'ús documentats a l'apartat anterior. Per a cadascun d'ells, es presenta:

- El diagrama de casos d'ús global del sistema.
- El diagrama de classes específic del cas d'ús.
- El diagrama de seqüència que mostra la interacció entre actors i objectes.

### 4.1 Diagrama classes general

El diagrama de casos d'ús representa les principals funcionalitats del sistema i la seva interacció amb els actors implicats: l'usuari (persona que utilitza el guant) i el desenvolupador (per a l'actualització del diccionari).

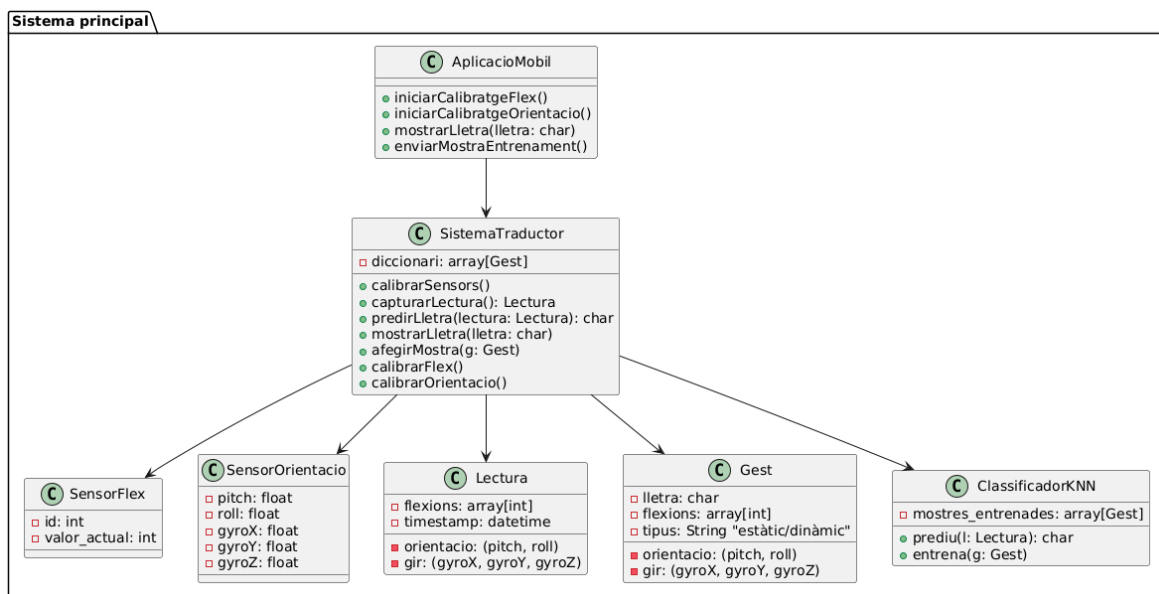


Figura 3. Diagrama de classes general

## 4.2 Anàlisi per cas d'ús

A continuació s'analitza cada cas d'ús de forma individual, amb els seus corresponents diagrames de classes i seqüència.

### 4.2.1 Calibratge del sensor d'orientació

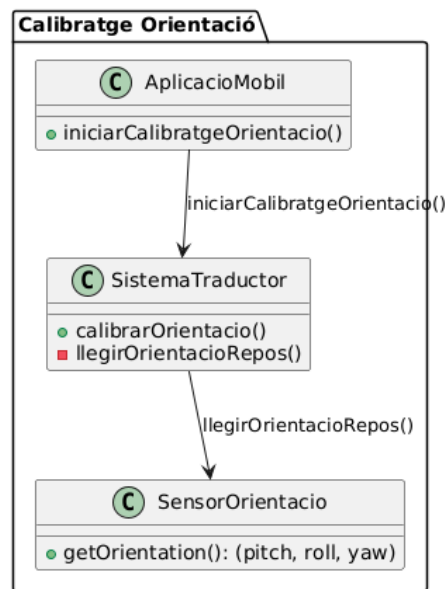


Figura 4. Diagrama de classes Cas d'ús 01

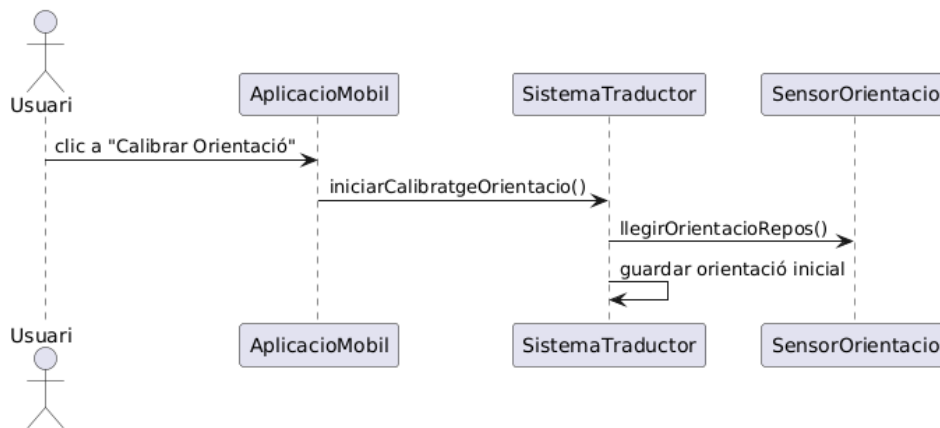


Figura 5. Diagrama de seqüències Cas d'ús 01

4.2.2 Calibratge dels sensors de flexibilitat

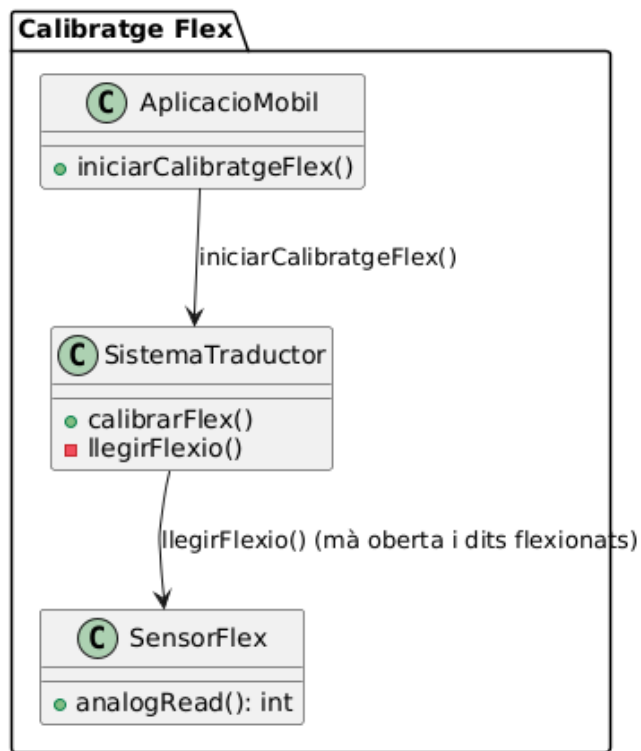


Figura 4. Diagrama de classes Cas d'ús 02

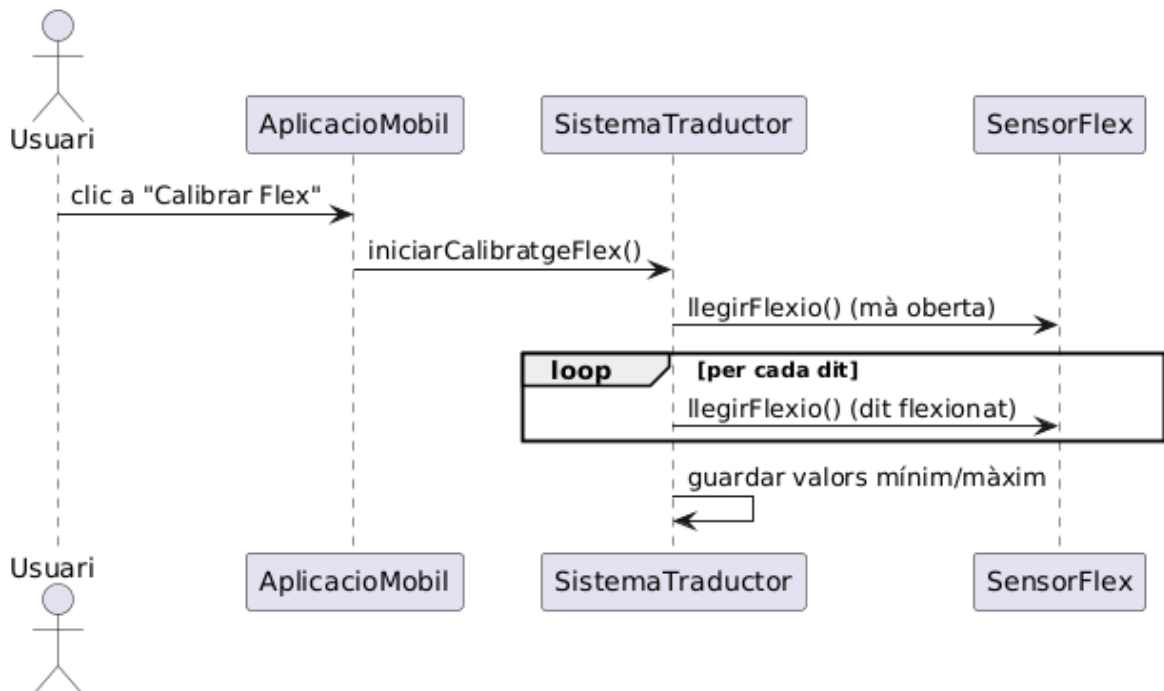


Figura 5. Diagrama de seqüències Cas d'ús 02

4.2.3 Captura de dades amb sensors

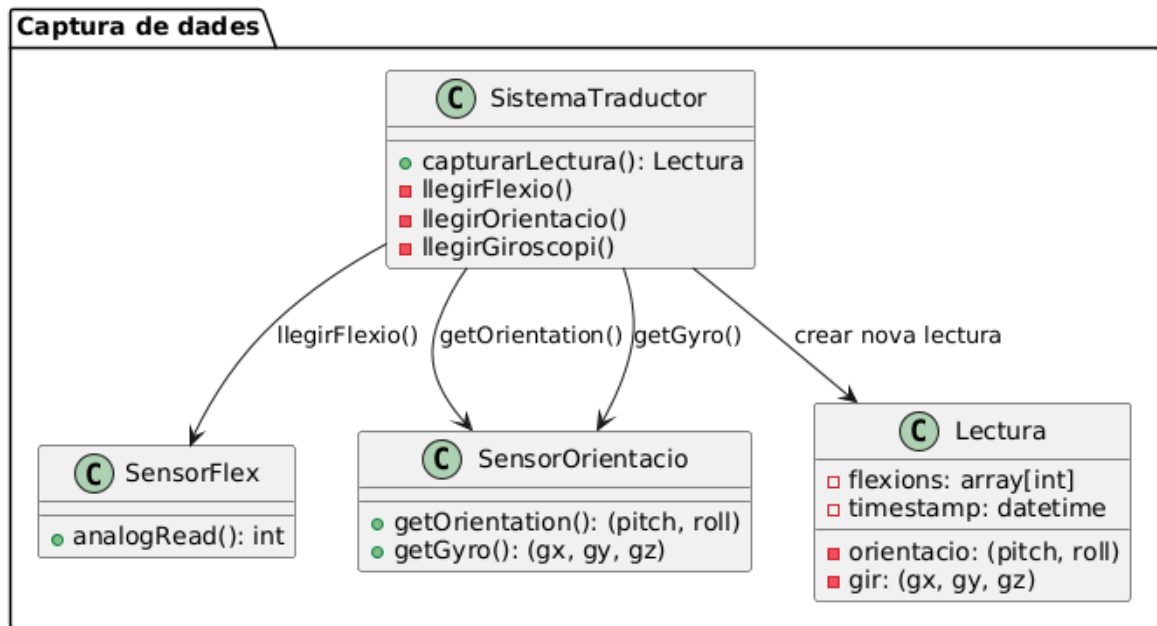


Figura 6. Diagrama de classes Cas d'ús 03

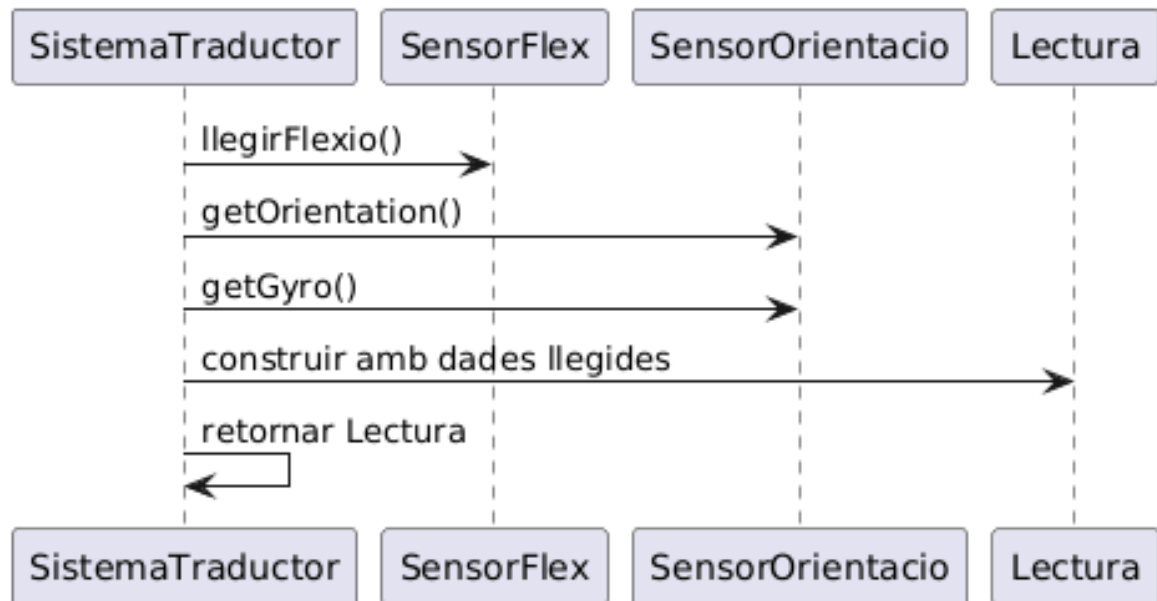


Figura 7. Diagrama de seqüències Cas d'ús 03

4.2.4 Predicció de la lletra amb K-NN

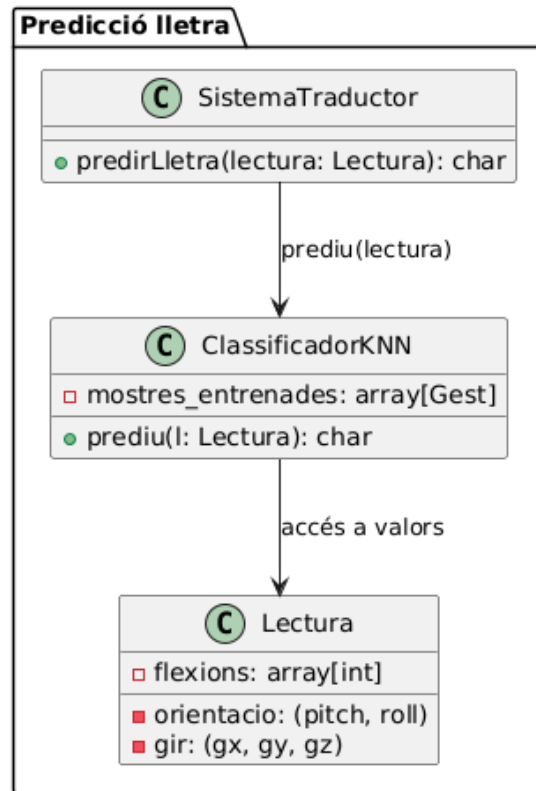


Figura 8. Diagrama de classes Cas d'ús 04

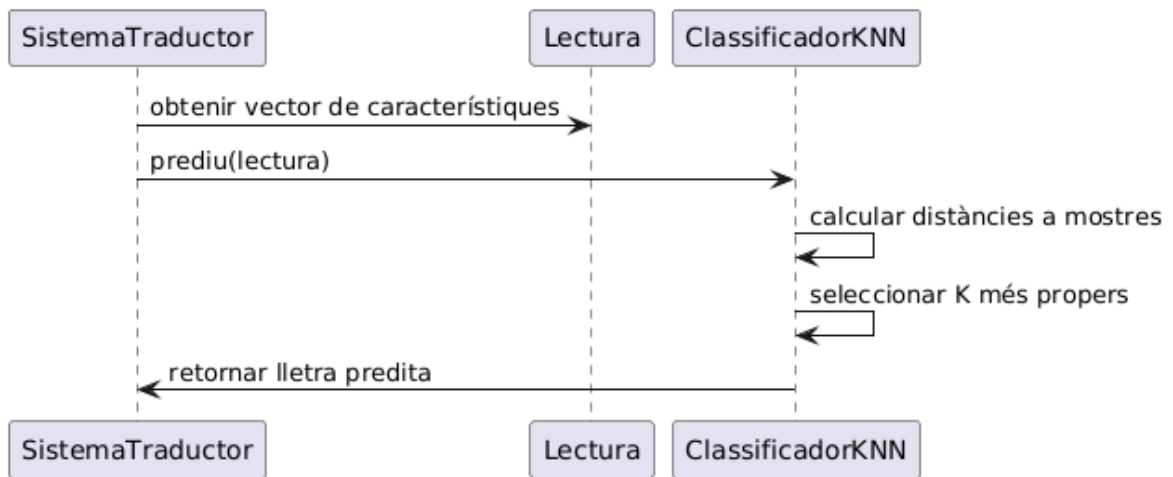


Figura 9. Diagrama de seqüències Cas d'ús 04

4.2.5 Visualització de la lletra

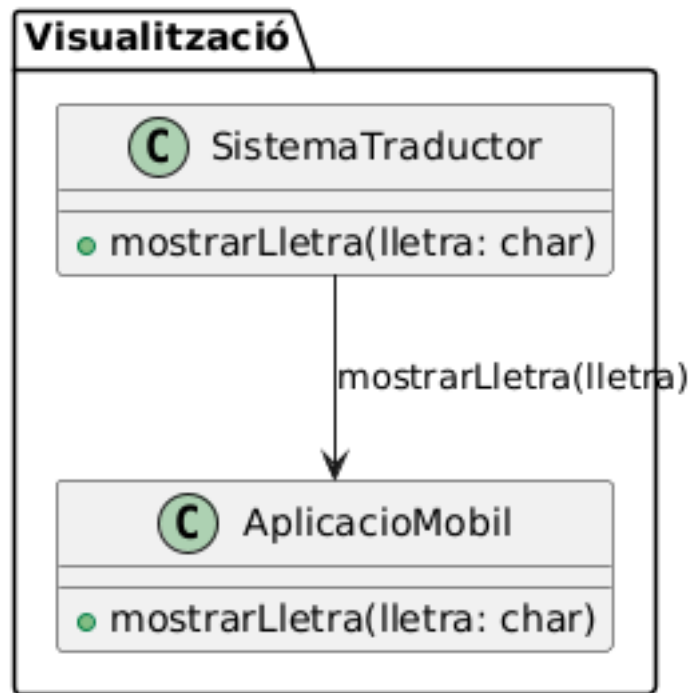


Figura 10. Diagrama de classes Cas d'ús 05

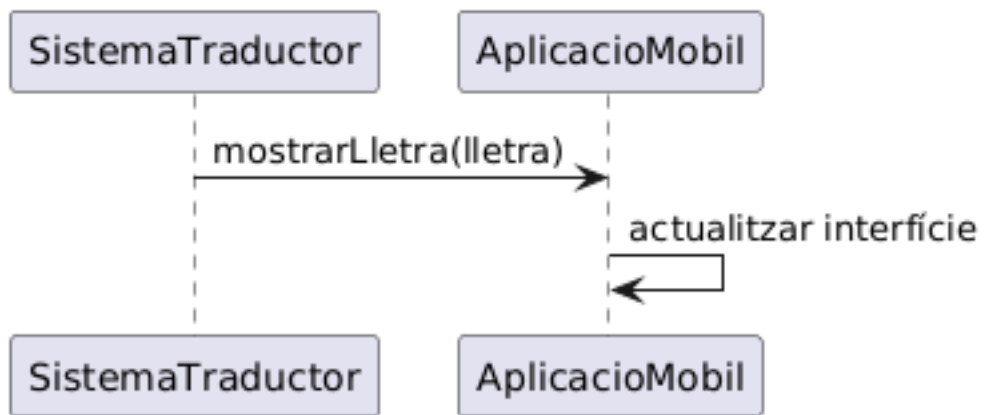


Figura 11. Diagrama de seqüències Cas d'ús 05

4.2.6 Entrenament del diccionari de gestos

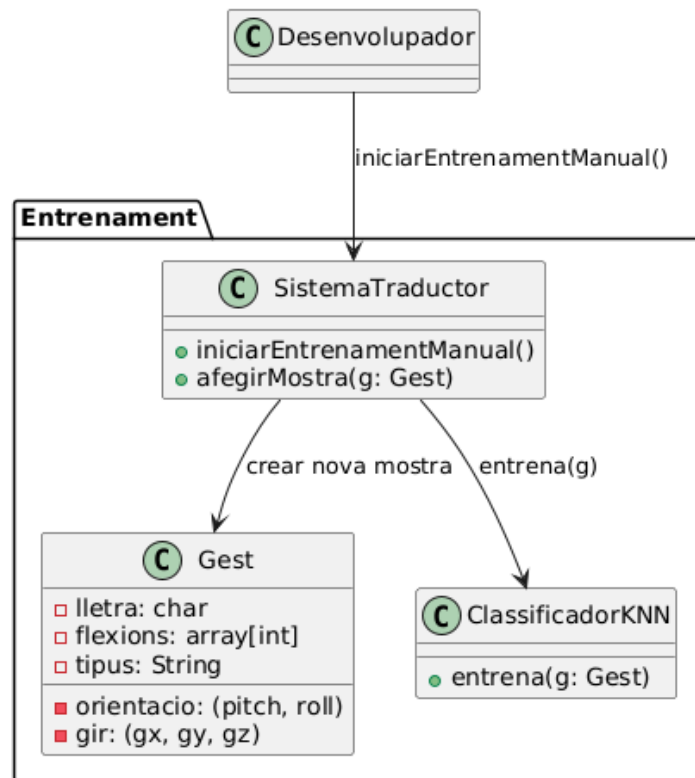


Figura 12. Diagrama de classes Cas d'ús 06

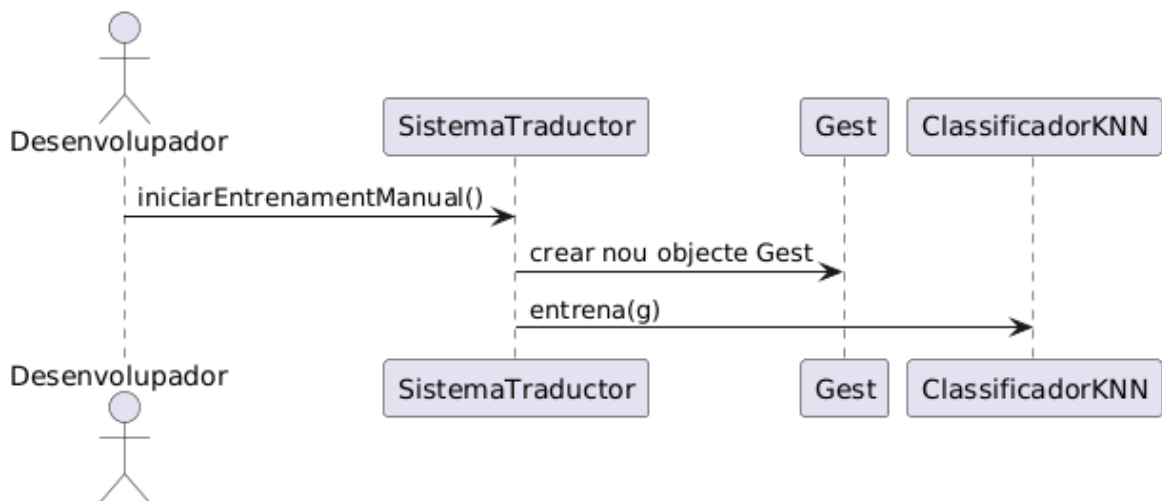


Figura 13. Diagrama de seqüències Cas d'ús 06

## 5 Disseny

En aquest apartat es presenta el disseny del sistema implementat. L'objectiu principal és descriure l'arquitectura general del projecte, la justificació en l'elecció de components, eines de desenvolupament, el seu esquema de connexió, l'organització de les seves funcions, la interfície amb l'usuari i la gestió de les dades internes. Tot el disseny s'ha concebut buscant simplicitat, eficiència i escalabilitat, amb la voluntat que el projecte pugui evolucionar en el futur sense necessitat de reestructuració profunda.

### 5.1 Arquitectura general del sistema

El sistema es pot dividir en tres blocs funcionals principals:

- Captació de dades: A través de sensors instal·lats al guant (sensor d'orientació i sensor de flexibilitat), s'obtenen dades en temps real dels moviments de la mà.
- Processament: Una placa microcontroladors centralitza les dades, les interpreta i aplica un algorisme de classificació per reconèixer la lletra realitzada.
- Visualització i interacció: La informació es transmet via Bluetooth a una aplicació mòbil, on l'usuari pot veure la lletra reconeguda, contruir paraules i fer calibratges.

### 5.2 Elecció dels components

L'elecció dels components que formen part del sistema ha estat una de les fases més rellevants del procés de disseny. Aquesta selecció s'ha fet amb la finalitat d'aconseguir un sistema robust, funcional i econòmic, capaç de complir els objectius del projecte sense requerir grans recursos. Els criteris principals que han guiat aquesta tria han estat la disponibilitat comercial, el cost assequible, la facilitat d'integració amb la resta del sistema i el suport per part de la comunitat de desenvolupadors.

Per capturar els moviments dels dits, es va optar per utilitzar sensors de flexibilitat de tipus resistiu. Inicialment es van considerar altres alternatives com ara sensors inercials individuals per a cada dit o fins i tot sensors capacitius, però aquests requerien una infraestructura de processament més complexa i no garantien una lectura estable en condicions de portabilitat. Els sensors resistius, en canvi, ofereixen una lectura directa basada en la variació de resistència en funció de la flexió del dit, la qual cosa permet integrar-los fàcilment al sistema a través d'un divisor de tensió<sup>1</sup>. A més, el seu cost és molt reduït i són més comuns de trobar en venda, la qual cosa facilita la seva adquisició.

Pel que fa a la detecció de l'orientació de la mà, es va seleccionar el sensor MPU-6050, el qual integra un acceleròmetre i un giroscopi de tres eixos. Aquesta combinació proporciona la informació necessària per detectar inclinació (pitch, roll i yaw) i moviments angulars (velocitat de gir) en temps real. Inicialment es va estudiar la possibilitat d'utilitzar el MPU-9250, una versió més avançada que afegeix una brúixola digital (magnetòmetre) per millorar la precisió en la determinació de l'orientació absoluta. Tanmateix, aquesta

---

<sup>1</sup> Un divisor de tensió és un circuit format habitualment per dues resistències en sèrie, utilitzat per reduir la tensió d'entrada fins a un valor desitjat. Es fa servir, per exemple, per adaptar la lectura d'un sensor a una entrada analògica d'un microcontrolador.

funcionalitat no era necessària per als objectius del projecte, ja que les lletres a reconèixer no depenen de la posició absoluta respecte als punts cardinals, sinó de la inclinació relativa de la mà i els moviments realitzats. A més, el MPU-6050 és més senzill de configurar, té una disponibilitat més gran en el mercat i una comunitat d'usuaris molt extensa amb abundant documentació i suport en forma de llibreries i tutorials. Això el converteix en una opció més pràctica, assequible i suficient per al tipus d'anàlisi que es pretén dur a terme.

La peça central de tot el sistema és la placa SparkFun Thing Plus ESP32-WROOM, basada en el microcontrolador ESP32. Aquesta elecció respon a diversos requisits essencials. En primer lloc, la necessitat d'una connexió sense fils, ja sigui Wi-Fi o Bluetooth, per poder comunicar-se amb un dispositiu mòbil sense cables. El mòdul ESP32 inclou de manera nativa Bluetooth clàssic (SPP<sup>2</sup>), que és precisament el protocol utilitzat per la connexió amb l'aplicació Android. En segon lloc, era indispensable disposar d'un nombre suficient de pins analògics per connectar els sensors de flexibilitat, concretament, cinc entrades analògiques independents, així com alimentació a 3.3 V (VCC), GND i línies de comunicació SCL (Serial Clock Line) i SDA (Serial Data Line) per integrar el sensor MPU-6050 via I<sup>2</sup>C<sup>3</sup>.

Tot i valorar altres opcions com l'Arduino Nano 33 IoT o la Raspberry Pi Pico, es van descartar per diferents motius:

- L'Arduino Nano 33 IoT compleix els requisits tècnics del projecte, com ara disposar de connexió Bluetooth, Wi-Fi, pins analògics suficients i compatibilitat amb els protocols I<sup>2</sup>C per connectar el sensor d'orientació, es va descartar principalment pel seu cost elevat en comparació amb l'ESP32. Dins d'un projecte que prioritza l'accessibilitat i l'eficiència econòmica, el preu més competitiu de l'ESP32 el va convertir en una opció més adequada sense sacrificar funcionalitats essencials.
- La Raspberry Pi Pico podia semblar una opció adequada pel fet d'incloure connexió Bluetooth i Wi-Fi, es va descartar principalment per la limitació del nombre de pins analògics disponibles, que resultava insuficient per llegir simultàniament els valors dels cinc sensors de flexibilitat.

Finalment, per al desenvolupament de la interfície d'usuari i la visualització de la informació reconeguda, es va optar per construir una aplicació per a dispositius mòbils Android. Aquesta decisió es justifica pel fet que Android és el sistema operatiu més estès a nivell mundial en l'àmbit dels smartphones, i ofereix un entorn de desenvolupament molt complet mitjançant Android Studio. La comunicació amb la placa ESP32 s'ha implementat a través de Bluetooth clàssic (SPP), i el llenguatge escollit per al desenvolupament ha estat Kotlin, per la seva integració nativa i les millores que ofereix respecte a Java en termes de simplicitat i llegibilitat del codi.

---

<sup>2</sup> El SPP (Serial Port Profile) és un perfil de comunicació Bluetooth que simula una connexió de port sèrie (UART) sense fils, que permet l'enviament de dades de manera continua entre dispositius.

<sup>3</sup> El bus I<sup>2</sup>C (Inter-Integrated Circuit) és un protocol de comunicació en sèrie que permet connectar diversos dispositius (sensors, microcontroladors, etc.) utilitzant només dues línies: una per a dades (SDA) i una per a senyal de rellotge (SCL).

Per a la programació de la placa ESP32, s'ha optat per PlatformIO integrat dins de Visual Studio Code en comptes de l'IDE d'Arduino clàssic. Aquesta decisió s'ha pres pel fet que PlatformIO ofereix un entorn de desenvolupament més robust, modern i escalable, amb suport per a múltiples plataformes i una millor gestió de llibreries i dependències. A més, permet treballar amb un sistema de projectes més estructurat, facilita el control de versions i ofereix una major flexibilitat a l'hora de compilar i desplegar el codi. Aquestes característiques han resultat molt útils en un projecte que requereix iteració contínua, gestió de múltiples mòduls i manteniment del codi a llarg termini.

En conjunt, la selecció dels components s'ha fet amb l'objectiu d'obtenir una arquitectura equilibrada, eficient i adaptada a les necessitats del projecte. Aquest conjunt d'eines i tecnologies no només permet assolir els objectius plantejats sinó que també obre la porta a futures ampliacions i millores del sistema.

### 5.3 Disseny de la interfície amb l'usuari

El disseny de la interfície d'usuari (UI) s'ha plantejat amb l'objectiu de maximitzar la usabilitat i facilitar la interacció amb el sistema de reconeixement de gestos a través del telèfon mòbil. En comptes d'utilitzar interfícies físiques com pantalles LCD o botons, tota la comunicació entre l'usuari i el sistema es realitza des de l'aplicació Android desenvolupada específicament per aquest projecte.

L'aplicació mòbil ha estat dissenyada seguint criteris de simplicitat i claredat visual, utilitzant components textuais i botons intuïtius. El flux principal d'interacció es basa en:

- Visualització en temps real de la lletra detectada pel sistema, amb un component TextView de gran mida per tindre una alta llegibilitat.
- Construcció progressiva de paraules i frases, on cada lletra reconeguda s'afegeix automàticament a una paraula en construcció, que a la vegada pot formar part d'una frase.
- Botons d'interacció ràpida, que permeten:
  - Afegir un espai per separar paraules.
  - Esborrar l'última lletra escrita.
  - Aplicar una suggerència predictiva si es detecta una paraula parcial que coincideix amb alguna del diccionari intern.

A més, la interfície incorpora una secció indispensable i dues seccions clau per l'ús òptim dels sensors:

- Connexió via Bluetooth: Un botó indispensable que gestiona la connexió entre l'aplicació mòbil i el microcontrolador indicant de forma visual quant la connexió s'ha realitzat correctament.
- Calibratge del sensor d'orientació: L'usuari col·loca la mà en posició neutral i prem un botó que fixa la referència inicial per al càlcul de la inclinació i rotació.
- Calibratge dels sensors de flexibilitat: L'usuari és guiat pas a pas per a realitzar correctament la seqüència de moviments necessaris per determinar els valors mínim i màxim de flexió de cada dit.

El resultat és una interfície adaptada a persones sense coneixements tècnics, que pot ser utilitzada de manera autònoma i intuïtiva. L'elecció d'una aplicació mòbil com a principal mitjà d'interacció permet una experiència portàtil, directa i visual, essencial per a un projecte orientat a la comunicació i l'accessibilitat.

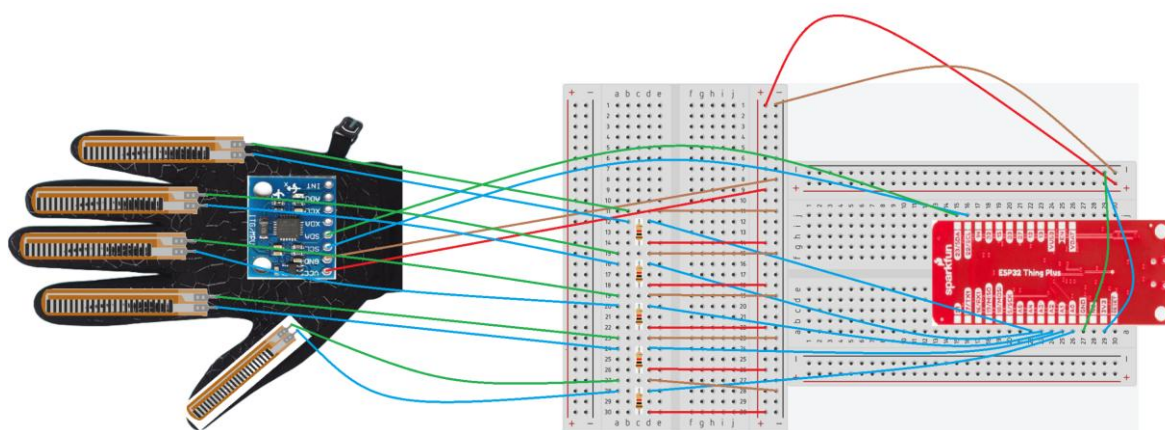
#### 5.4 Esquema de connexions electròniques

Per garantir el funcionament correcte del sistema i la integració de tots els components físics, s'ha dissenyat un esquema de connexions que mostra detalladament la disposició dels sensors sobre el guant, la interconnexió amb el sensor d'orientació MPU-6050 i la connexió amb la placa de control ESP32 mitjançant una protoboard<sup>4</sup>.

Els cinc sensors de flexibilitat estan connectats a cinc entrades analògiques independents de la placa ESP32. Cadascun d'ells es connecta a través d'un divisor de tensió amb resistències de valor adequat, per garantir una lectura estable i proporcional al grau de flexió del dit. Aquestes connexions s'estableixen a través d'una protoboard per facilitar la distribució d'alimentació i la disposició ordenada dels senyals.

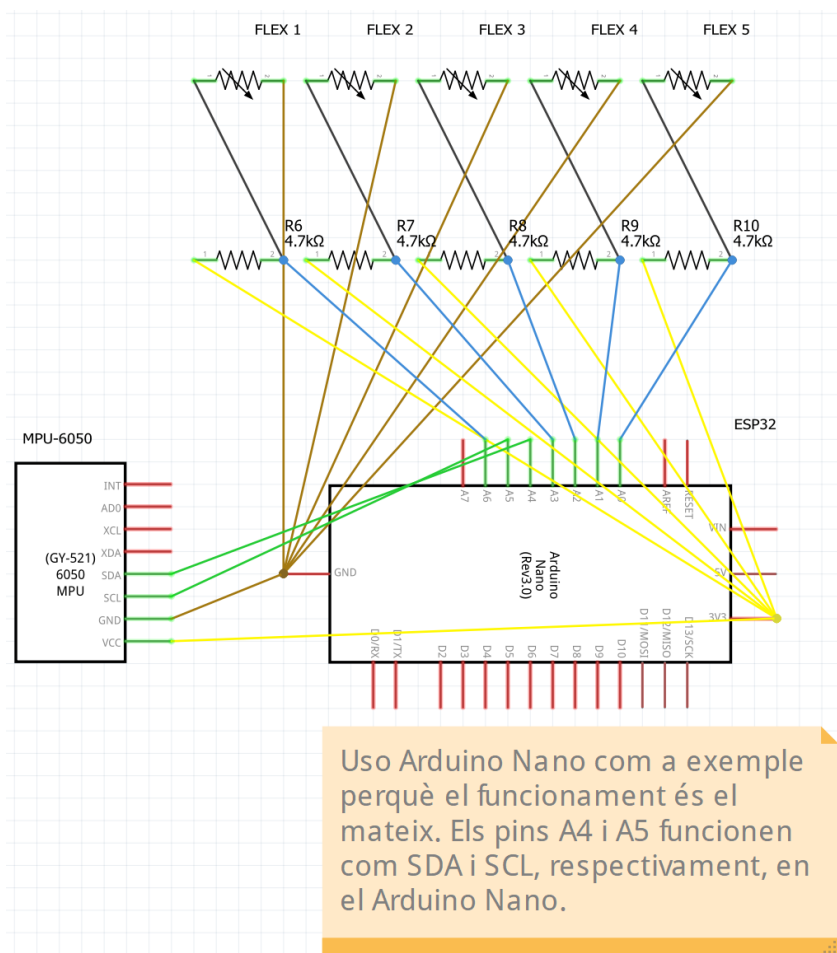
El sensor MPU-6050 es comunica amb la placa ESP32 mitjançant interfície I<sup>2</sup>C, utilitzant les línies SDA i SCL, a més de connexions per a alimentació (VCC, 3.3 V) i GND. Aquesta connexió permet capturar en temps real les dades d'orientació i rotació angular de la mà de l'usuari.

L'alimentació general del sistema es fa mitjançant el port USB de la placa ESP32, que també pot alimentar la protoboard a través dels rails d'alimentació positius i negatius.



**Figura 14.** Esquema detallat de connexions entre els sensors, la placa ESP32 i la protoboard

<sup>4</sup> Una protoboard (o placa de prototipatge) és una eina que permet connectar components electrònics entre si de manera temporal, sense necessitat de soldadures. S'utilitza habitualment per al desenvolupament i test de circuits electrònics de manera ràpida i reutilitzable.



- Valors de calibratge, que inclouen els màxims i mínims de flexibilitat de cada dit i la posició inicial d'orientació del sensor MPU6050.

## 5.6 Estratègia de comparació de gestos

El sistema es basa en la classificació de gestos mitjançant un algorisme K-Nearest Neighbors (K-NN). Aquesta tècnica ha estat escollida per la seva simplicitat, facilitat d'implementació i bon comportament en sistemes amb poca potència computacional com l'ESP32.

Cada gest entrenat s'emmagatzema com un vector de característiques compost per:

- Valors de flexió de cada dit (després de calibració).
- Valors d'orientació (pitch, roll i yaw) respecte la posició inicial.
- Valors de velocitat de gir (per als gestos dinàmics).

Quan l'usuari realitza un nou gest, el sistema recull una nova lectura i calcula la distància euclidiana respecte a totes les mostres prèviament entrenades. Es seleccionen els k gestos més propers i, per votació majoritària, s'obté la lletra més probable.

Aquest enfocament ha permès evitar l'ús de xarxes neuronals o classificadors més complexos, els quals requeririen més capacitat de càlcul i memòria. A més, el sistema és completament adaptable, ja que l'usuari pot tornar a entrenar noves mostres i actualitzar la base de dades del K-NN de manera dinàmica i personalitzada.

També s'ha implementat una distinció entre gestos estàtics i dinàmics, fent servir les lectures de rotació angular (gyro) per detectar moviments. Això permet millorar la precisió en la identificació de lletres com "H", "J" o "X" que impliquen moviment.

## 5.7 Conclusió

El disseny del guant traductor de llenguatge de signes ha estat orientat des del principi cap a la simplicitat estructural, la modularitat del codi i la viabilitat tècnica amb recursos limitats. La separació clara de responsabilitats entre les diferents parts del sistema (captura de dades, processament i sortida) ha permès construir una base sòlida i fàcilment escalable.

La decisió de mantenir una interfície mínima basada en port sèrie, així com l'ús d'una estratègia de comparació directa per a la detecció de gestos, ha estat clau per garantir una primera versió funcional, estable i amb possibilitats d'evolució. Aquest enfocament ofereix una plataforma des de la qual es podria, en futures iteracions, integrar millores com l'aprenentatge automàtic, la connexió amb aplicacions mòbils o l'expansió a vocabularis més amplis.

Amb aquest disseny, el projecte no només demostra la seva viabilitat tècnica, sinó també la seva capacitat d'adaptació a nous reptes i escenaris d'ús.

## 6 Implementació

En aquest apartat es descriuen, de manera estrictament tècnica, els components de maquinari i les estructures de programació utilitzades en el desenvolupament del guant traductor de llengua de signes, així com els mètodes de lectura, processament i classificació de les dades capturades.

### 6.1 Entorn de desenvolupament

Per al desenvolupament del programari del guant traductor de llenguatge de signes s'ha utilitzat Visual Studio Code juntament amb l'extensió PlatformIO, un entorn de desenvolupament modern i versàtil orientat al desenvolupament encastat.

PlatformIO permet:

- La configuració de pins, l'ús de diferents entorns de compilació i la gestió de llibreries de forma modular.
- Compilació i pujada eficient del firmware<sup>5</sup>.
- Control de versions i escalabilitat del projecte.

El projecte es basa en la lectura contínua de dades dels sensors connectats al guant:

- Cinc sensors de flexibilitat (Flex Sensors), connectats a les entrades analògiques.
- Un sensor d'orientació MPU6050, connectat via bus I2C.

La lògica del programa principal s'executa dins d'un bucle infinit, on es processen les dades recollides i es comparen amb un diccionari intern per detectar la lletra corresponent del llenguatge de signes.

A continuació es mostra un fragment representatiu de la lògica principal:

```
void loop() {
  llegirFlexSensors();
  llegirOrientacio();
  detectarLletra();
  mostrarLletra();
  delay(100);
}
```

Aquest cicle garanteix una lectura continua de dades i una actualització constant de la predicció.

---

<sup>5</sup> El firmware és el conjunt d'instruccions programades que controlen el funcionament intern d'un dispositiu electrònic. En aquest projecte es tracta del codi que s'executa al microcontrolador ESP32.

### 6.1.1 Configuració de PlatformIO i llibreries utilitzades

Per al desenvolupament del guant traductor de llenguatge de signes, s'han utilitzat diverses llibreries de codi obert per facilitar la comunicació amb els sensors i estructurar el codi del projecte de manera clara i eficient.

Llibreria	Descripció
Adafruit MPU6050 (v2.2.6)	Permet la lectura de dades del sensor d'orientació MPU6050 via I2C. Inclou suport per acceleròmetre i giroscopi, i es basa en la Adafruit Unified Sensor
MPU6050_light (v1.1.0)	Ofereix inicialització ràpida i càlcul d'angles de rotació amb menor consum de recursos
Wire (inclosa amb el framework Arduino)	Llibreria estàndard per a la comunicació I2C, necessària per a ambdues biblioteques anteriors

Taula 1. Llibreries usades i les seves descripcions

## 6.2 SparkFun Thing Plus ESP32 WROOM-32E

L'SparkFun Thing Plus ESP32 WROOM és la principal placa de control d'aquest projecte. És una placa econòmica basada en el microcontrolador ESP32 d'Espressif amb versatilitat i connectivitat sense fils (Wi-Fi i Bluetooth).



Figura 16. Imatge d'una SparkFun Thing Plus ESP32 WROOM-32E [2]

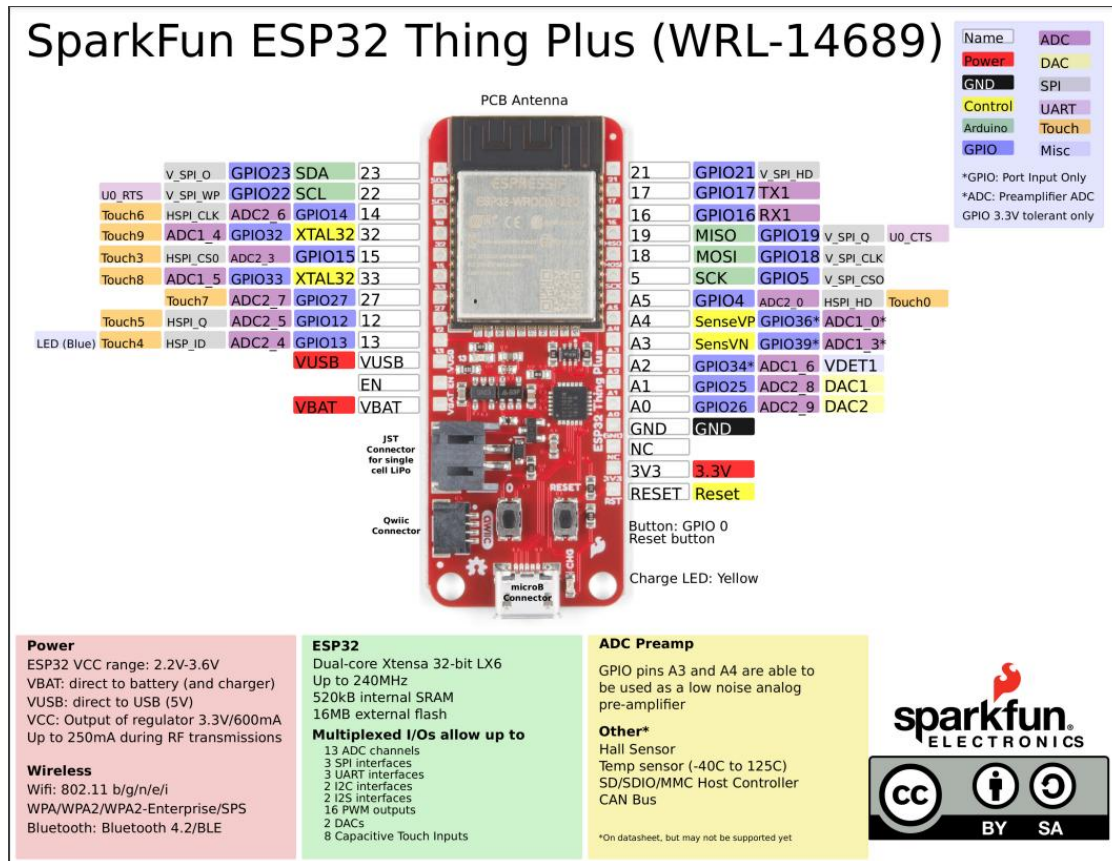


Figura 17. Esquema de pins del ESP32 [3]

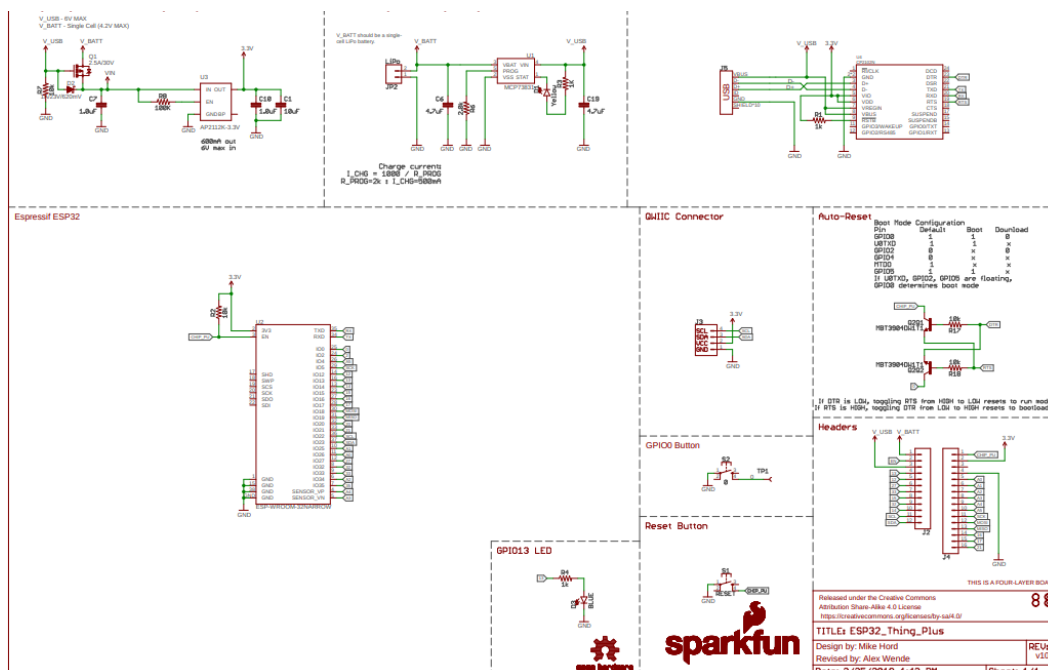


Figura 18. Esquema del ESP32 [4]

### 6.2.1 *Descripció funcional*

Microcontrolador: ESP32 de doble nucli, freqüència de rellotge de fins a 240 MHz.

CPU i Memòria interna:

- Microprocessador de freqüència fins a 240 MHz
- 520 KB de SRAM
- 16 KB de SRAM en RTC, 8 KB de RTC FAST (accedida per la CPU principal durant el RTC Boot des del Deep-sleep mode, 8 KB de RTC SLOW (accedida pel coprocessador durant el Deep-sleep mode
- 448 KB ROM

Connectivitat:

- Wi-Fi 802.11 b/g/n i Bluetooth 4.2 BR/EDR
- Bit rate de 802.11n fins a 150Mbps

Alimentació i voltatge d'operació:

- 3,3 V.
- Es pot alimentar a través del port USB o mitjançant el pin VBAT.

Entrades/Sortides (GPIO): Ofereix múltiples pins digitals i analògics, a més de busos de comunicació com UART, I2C i SPI.

### 6.2.2 *Integració al projecte*

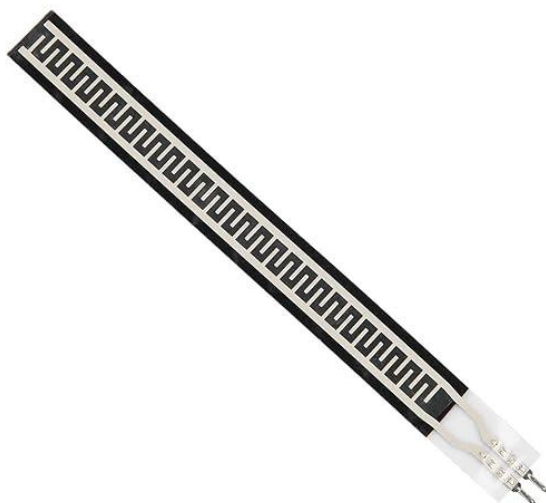
L'ESP32 s'encarrega de les següents tasques dins del guant traductor:

- Llegir les dades dels cinc sensors de flexibilitat i del sensor MPU6050.
- Comparar les lectures amb el diccionari intern de gestos.
- Mostrar per port sèrie la lletra detectada i valors dels sensors.
- Permetre calibratge i ampliació futura del diccionari de gestos.

## 6.3 **Sensors Flex**

Els sensors de flexibilitat són un dels components fonamentals del sistema, ja que permeten detectar la postura de cada dit de la mà. Gràcies a ells, el guant és capaç d'interpretar diferents lletres del llenguatge de signes basant-se en la configuració dels dits.

Cada sensor es basa en el principi de la resistència variable: quan el sensor es manté recte, la seva resistència interna és baixa; quan es doblega, la resistència augmenta proporcionalment a l'angle de flexió. Aquesta propietat permet mesurar de forma indirecta la curvatura de cada dit.



**Figura 19.** Sensor de flexibilitat [5]

### ***6.3.1 Descripció funcional***

Principi de funcionament: Basat de resistència variable; quan el sensor es doblega, la seva resistència interna augmenta proporcionalment.

Dimensions:

- Longitud de 100 mm/3,9 polzades
- Amplada de 10 mm/0,4 polzades
- 2 pins de distància 2,54 mm/0,1 polzades

Especificacions:

- Temps de resposta de 10 ms i recuperació de 15 ms
- Voltatge d'operació de 3,3 V
- Rang de mesura de 0-500g

### ***6.3.2 Integració al projecte***

Cada sensor s'ha connectat a una entrada analògica de l'ESP32. El valor llegit per cada sensor correspon a una lectura ADC (Analog to Digital Converter) proporcional a la resistència del sensor, que depèn de la flexió del dit.

La lectura s'obté mitjançant la funció `analogRead()`, que retorna un valor digital entre 0 i 4095 (resolució de 12 bits de l'ESP32).

El sistema utilitza cinc sensors, un per a cada dit de la mà, que permeten obtenir una imatge instantània de la postura de la mà en cada moment.

## 6.4 Sensor d'orientació MPU-6050

El MPU-6050 és un sensor de moviment que integra un acceleròmetre de 3 eixos i un giroscopi de 3 eixos en un sol xip, amb una resolució de 16 bits per canal (Naylamp Mechatronics, s.d.) [6]. És àmpliament utilitzat per detectar orientació, inclinació i moviment en sistemes encastats i dispositius portables.

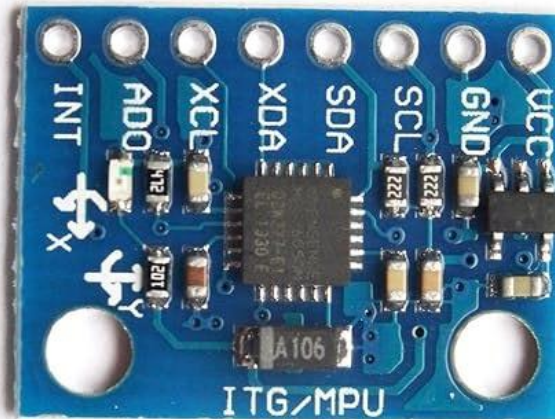


Figura 20. Sensor d'orientació MPU-6050 [7]

### 6.4.1 Descripció funcional

Components interns:

- Acceleròmetre 3D: Rang configurable (fins a  $\pm 16g$ ). Per defecte:  $\pm 2g$
- Giroscopi: Rang configurable (fins a  $\pm 2000^\circ/s$ ). Per defecte:  $\pm 250^\circ/s$
- DMP (Digital Motion Processor) per processar les dades internes

Tensió d'operació: 3,3 V

Dimensions: 78 x 50 x 11 mm; 20 g

Rendiment:

- Freqüència d'actualització fins a 1000 Hz
- Conversió analògic-digital interna de 16 bits per cada canal

Mode de comunicació: Bus I2C estàndard

### 6.4.2 Lectura de dades i conversió

El sensor retorna valors digitals de 16 bits amb signe (rang de  $-32768$  a  $32767$ ), que representen:

- L'acceleració en cada eix (X, Y, Z)

- La velocitat angular en cada eix (X,Y,Z)

Per convertir aquestes lectures a unitats físiques s'utilitzen les fórmules següents [8]:

Acceleració (en g):

$$\text{Accel} = (\text{Valor brut}/32768) * \text{Rang acceleròmetre} \quad (1)$$

Velocitat angular (en °/s):

$$\text{Gyro} = (\text{Valor brut}/32768) * \text{Rang giroscopi} \quad (2)$$

Aquesta conversió només és necessària si es volen obtenir valors físics exactes. En aquest projecte, però, s'ha optat per treballar directament amb els valors bruts, tal com es justifica a continuació.

#### 6.4.3 Ús de `getMotion6()` i `int16_t`

En aquest projecte s'ha utilitzat la funció `getMotion6()` de la llibreria del MPU6050. Aquesta funció llegeix simultàniament els sis valors del sensor (acceleròmetre i giroscopi) i els desa en variables de tipus `int16_t`, corresponents al format de sortida natiu del sensor.

Aquesta decisió es basa en diversos motius tècnics:

- Eficiència computacional: El processament amb enters és molt més ràpid i lleuger que amb nombres de punt flotant, especialment en microcontroladors com l'ESP32.
- Simplicitat en la comparació: El projecte no necessita calcular valors físics, sinó comparar patrons de gestos (valors bruts) amb una base de dades definida. Això elimina la necessitat de conversió i redueix el càlcul.
- Adequat per a l'objectiu: Com que només s'ha de reconèixer si un gest correspon a una lletra, no cal saber exactament l'acceleració o la velocitat angular en unitats físiques.

Aquest enfocament ha permès mantenir el codi net, eficient i fàcil de modificar. L'ús de `getMotion6()` simplifica la lectura de dades i evita càlculs innecessaris, mantenint l'enfocament del projecte en la detecció de patrons i no en la mesura precisa del moviment.

## 7 Procés iteratiu de desenvolupament i millora

El procés d'implementació d'aquest projecte no ha estat lineal, sinó iteratiu i centrat en la prova i l'error, partint de conceptes bàsics fins a aconseguir una versió funcional, escalable i estable del guant traductor. A continuació es detalla tot el procediment realitzat, les dificultats trobades i les solucions aplicades.

### 7.1 Lectura inicial dels sensors de flexibilitat

Els sensors de flexibilitat utilitzats en aquest projecte es basen en el principi de resistència variable: quan el sensor es doblega, la seva resistència interna augmenta proporcionalment a l'angle de flexió. A causa d'aquesta naturalesa resistiva, no és possible obtenir directament una lectura digital del seu estat.

Per tal de poder mesurar la flexió de manera efectiva, es va implementar un circuit divisor de tensió. Aquest circuit permet convertir els canvis de resistència en variacions de tensió, les quals són llegides pels pins analògics (ADC) del microcontrolador ESP32. Gràcies a la resolució de 12 bits dels ADC d'aquest microcontrolador, es van obtenir valors digitals compresos entre 0 i 4095.

Un cop realitzada la configuració dels pins i validada la lectura de dades, es van començar a observar valors de sortida que variaven de manera coherent amb el grau de flexió de cada dit. Aquesta observació va permetre entendre el comportament dels sensors i establir les bases per al processament de gestos.

### 7.2 Fixació dels sensors i proves inicials

Abans de desenvolupar cap lògica complexa, va ser necessari definir una manera de fixar els sensors al guant. Inicialment es va utilitzar cinta adhesiva de doble cara com una solució provisional que permetia validar el comportament dels sensors i fer les primeres proves per identificar les diferències dels gestos.

Amb els sensors col·locats, es va dissenyar un primer programa que comparava les lectures amb uns llindars hardcodedjats<sup>6</sup> (mínim i màxim per cada dit) per detectar si s'estava realitzant una lletra específica. Tot i que el sistema funcionava, va demostrar-se molt limitat, poc fiable i gens escalable.

Aquestes limitacions van motivar el desenvolupament d'un sistema de calibratge dinàmic.

---

<sup>6</sup> Valors que han estat escrits directament dins del codi font, en lloc de ser inicialitzats de forma dinàmica

### **7.3 Implementació d'un sistema de calibratge i millora en fixació**

Un dels problemes que es va trobar va ser que els valors dels sensors de flexibilitat no eren coherents entre proves, ja que podien transmetre valors diferents entre una prova i una altra per causes com:

- No doblar els dits de la mateixa manera que en les anteriors proves
- Una altra persona esta realitzant les proves amb el guant

Per solucionar l'estabilitat es van cosir unes cintes elàstiques als dits del guant per poder usar-les com a fundes i passar per dins els sensors de flexibilitat.

Per tal de millorar la precisió i adaptar el sistema a cada mà, es va desenvolupar un sistema de calibratge dinàmic. Aquest sistema demanava a l'usuari que deixi la mà oberta amb els dits estirats i seguidament tancar la mà amb els dits flexionats. Així el programa podia establir els valors mínims i màxims que podia assolir cada sensor segons la flexió dels dits. Aquests valors es van canviar per tindre un rang entre 0 i 1024 en compte de 0 i 4095 per evitar variacions lleugeres quan es flexionen els dits.

A més, es va desenvolupar un sistema de calibratge dinàmic. Aquest sistema demanava a l'usuari que realitzés una sèrie de gestos per a cada lletra, i a partir de les lectures, fixava automàticament un marge vàlid de valors.

Aquest calibratge va suposar una millora substancial, especialment en les lletres estàtiques, ja que permetia personalitzar el reconeixement segons l'usuari.

### **7.4 Integració del sensor d'orientació**

Amb el sistema de flexibilitat parcialment implementat, es va incorporar el sensor d'orientació MPU6050, el qual permet obtenir valors d'inclinació en els eixos x, y i z (pitch, roll, yaw), i valors per a la velocitat de gir (gyroscope). Inicialment només es van utilitzar els valors d'inclinació per poder diferenciar les lletres estàtiques que usen els mateixos valors de flexió, però, tenen diferent orientació, com M i P o N i U.

Per a una lectura més fiable, el sensor es va col·locar al centre del dors de la mà, fixat amb cinta adhesiva de doble cara per garantir la rigidesa durant els gestos.

Una vegada estava implantada la lògica del funcionament del sensor d'orientació MPU6050 es va implementar una calibració adicional on el sistema demanava a l'usuari que deixes la mà en una posició de repòs per poder detectar la posició inicial de la qual haurà de mirar l'orientació.

### **7.5 Detecció de lletres amb moviment i el cas especial de la Y**

Per a lletres amb moviment (com J, V o algunes variants gestuals), es va intentar detectar canvis sobtats en la velocitat angular. El sistema comparava les lectures de gyro per veure si s'havia fet un moviment ràpid. Aquest enfoc, però, no era estable: sovint produïa falsos positius o perdia moviments reals.

També es va trobar el cas especial de la lletra Y, que no implica orientació ni moviment angular, però sí una acció repetitiva de flexionar i estirar el dit petit. Per solucionar-ho es va crear una lògica específica que controla la variació de la flexió del dit petit dins d'un interval de temps, permetent detectar aquest patró rítmic.

## 7.6 Implementació d'un sistema K-NN per millorar la detecció

Al causa de la complexitat i inestabilitat del sistema basat en marges, es va decidir canviar per complet el enfocament del programa, i així, es va implementar un sistema més robust mitjançant K-Nearest Neighbors (K-NN), proposat per Cover i Hart (1967) [9], basat en distàncies euclidianes. Aquest sistema permet entrenar el reconeixement de lletres a partir de múltiples mostres per lletra (entre 3 i 5) que podrà fer l'usuari, emmagatzemades com vectors de referència.

Es van definir dos tipus de sessions d'entrenament:

- Lletres estàtiques: On s'utilitzen solament valors de flexió i inclinació.
- Lletres dinàmiques: On s'afegeixen també els valors de velocitat angular.

Per entrenar lletres dinàmiques, el sistema dona a l'usuari un interval de temps per realitzar el moviment, i després guarda el patró com una seqüència mitjana.

En temps d'execució, quan el sistema detecta un gest, compara el gest amb les mostres entrenades usant una funció de distància euclidiana i retorna la lletra més similar.

## 7.7 Funcionament del sistema K-NN

El sistema K-NN dins del projecte segueix els següents passos:

- Captura de lectures: El sistema recull les dades dels sensors implicats (flexió dels dits, orientació i, si es tracta d'una lletra dinàmica, velocitat angular).
- Construcció del vector de característiques: S'organitzen les lectures en un vector per poder-les usar més endavant a l'hora de fer la predicció de la lletra.
- Comparació amb les mostres entrenades: Es compara el vector amb les mostres prèviament recollides i guardades per a cada lletra.
- Càlcul de distàncies: Es calcula la distància euclidiana entre el vector capturat i cada mostra entrenada.
- Classificació final: Es seleccionen les K mostres més properes i es fa una votació majoritària per a determinar la lletra predita.

## 7.8 Millora de la fiabilitat i resultats obtinguts

Aquest nou enfocament usant el sistema de K-NN per detectar les lletres va representar un canvi radical en la fiabilitat del sistema. Les proves mostraven que, després de l'entrenament, es podia detectar sense quasi bé cap dificultat totes les lletres estàtiques i dinàmiques.

Aquesta millora ha fet que el sistema pugui ser considerat útil per a entorns reals i estableix una base sòlida per a versions futures més sofisticades i elaborades, amb classificadors més avançats.

## 7.9 Aplicació mòbil per a dispositius Android

Per fer el sistema més accessible i portàtil, s'ha desenvolupat una aplicació mòbil per a dispositius Android que permet a l'usuari veure el resultat del gestos que fa a través del telèfon mòbil que es connecta per Bluetooth a la placa ESP32. Aquesta aplicació complementa el funcionament del guant traductor, facilitant la calibració, la construcció de paraules i frases i la interacció amb el sistema d'una forma intuïtiva i funcional, sense necessitat de dependre d'un ordinador.

### 7.9.1 Funcionalitats principals

L'aplicació permet dur a terme les següents accions:

- Connexió Bluetooth amb l'ESP32: L'usuari pot vincular-se amb el microcontrolador a través d'un canal serial Bluetooth clàssic (SPP).
- Calibració dels sensors: Des de la interfície de l'app, es poden iniciar dues rutines de calibratge:
  - Calibració dels sensors de flexibilitat: L'usuari segueix les instruccions que li surten per pantalla. S'envia una comanda via Bluetooth, i el microcontrolador inicia la rutina per obtenir màxims i mínims de cada dit.
  - Calibració del sensor d'orientació (MPU6050): L'usuari col·loca la mà en posició neutral i es registren les referències de pitch, roll i yaw per establir una referència de l'inclinació i orientació angular de la posició inicial.
- Visualització i construcció de missatges: Inicialment, l'aplicació mostrava la lletra reconeguda en temps real que el sistema enviava per Bluetooth una vegada calibrat el guant. Més tard es van afegir noves funcionalitats que milloren significativament la usabilitat:
  - Creació de paraules: Cada lletra reconeguda s'afegeix automàticament a una paraula en construcció sempre que s'estigui fent el gest de la lletra durant un cert període de temps. Això permet formar paraules completes a partir de gestos successius.
  - Espai i frases: S'ha implementat un botó per afegir un espai de manera que les paraules s'afegeixen a una frase completa situada just a sota. Aquesta funcionalitat permet generar missatges sencers, millorant l'experiència de comunicació.
  - Esborrar l'última lletra: En cas d'error, l'usuari pot eliminar l'última lletra escrita prement un botó específic.

- Suggestiments de paraules: Per reduir el temps necessari per escriure paraules lletra per lletra, s'ha afegit un diccionari en format JSON (JavaScript Object Notation) amb grups de paraules associades a cada lletra inicial. Quan l'usuari comença a escriure, l'aplicació mostra una llista de paraules suggerides que coincideixen amb les lletres introduïdes. Aquesta llista s'actualitza dinàmicament a mesura que es van afegint noves lletres.
- Auto completat: Si l'usuari prem sobre el botó de suggeriment, l'aplicació omple automàticament la paraula en curs amb la suggerida (primera en ordre alfabètic en cas d'haver més d'una paraula suggerida), agilitant així la comunicació.

### 7.9.2 Desenvolupament tècnic

L'aplicació ha estat desenvolupada per a la plataforma Android utilitzant Android Studio. Aquesta aplicació actua com una interfície senzilla però funcional que es comunica directament amb la placa ESP32 mitjançant un canal serial Bluetooth.

Característiques tècniques:

- Llenguatge i entorn: Kotlin a Android Studio.
- Connexió: Serial Bluetooth a través de la classe BluetoothSocket amb UUIDs SPP.
- Estructures:
  - Handler: per gestionar els missatges entrants del buffer.
  - TextView: per mostrar la lletra reconeguda en temps real.
  - Button: per iniciar les rutines de calibratge.
  - ListView: mostra les paraules suggerides a partir del diccionari JSON
- Gestió del diccionari:
  - Diccionari carregat des de fitxer JSON local
  - Filtrat dinàmic de paraules segons les lletres escrites.
  - Integració del botó de suggeriment per aplicar el completat automàtic.
- Gestió de paquets:
  - Implementació d'un buffer circular per evitar problemes de latència o pèrdua de dades, optimitzant la recepció contínua de caràcters.
  - Filtrat i actualització constant de la lletra mostrada i dels continguts de la paraula/frase

### 7.10 Millores possibles

Tot i que el sistema desenvolupat presenta una funcionalitat sòlida i operativa per al reconeixement de gestos de la llengua de signes, existeixen diverses línies de millora que podrien incrementar-ne la precisió, la robustesa i la versatilitat:

- Millora en la fixació dels sensors de flexibilitat: Tot i que els sensors actuals estan integrats dins de cintes elàstiques cosides als dits del guant, es podria dissenyar una versió més estable mitjançant impressió 3D o materials més rígids que evitin moviments laterals o desviacions en la lectura de dades, garantint una col·locació idèntica entre usos.

- Incorporació de nous sensors de posició absoluta: L'ús de sensors com magnetòmetres o unitats de mesura inercial de 9 eixos (ex. MPU-9250 o BNO055) permetria detectar amb més precisió la rotació absoluta de la mà, millorant el reconeixement de lletres amb gestos dinàmics similars.
- Expansió del sistema de classificació: Tot i que l'algorisme K-NN ha estat eficaç, la incorporació de models més avançats com xarxes neuronals lleugeres (TinyML) o SVM podria millorar la classificació, especialment en entorns amb soroll o amb variabilitat en els gestos.
- Persistència de dades al dispositiu: El sistema actual no guarda històrics de sessions ni resultats. L'addició d'un sistema de memòria EEPROM o targeta SD permetria registrar sessions per a anàlisi posterior o entrenament incremental.
- Millores en la interfície d'usuari de l'aplicació mòbil: Actualment, la interfície ofereix suggeriments de paraules i opcions de correcció. Es podria afegir reconeixement de frases, historial de converses, opció de dictat per veu, o integració amb sistemes TTS (Text-To-Speech) per verbalitzar el text interpretat.
- Sistema multilinguatge i configuracions personalitzades: S'hi podria afegir suport multilinguatge o la possibilitat que l'usuari creï el seu propi diccionari de gestos personalitzat, ampliant així l'ús del guant a altres sistemes de signes o aplicacions específiques.
- Portabilitat completa mitjançant incorporació de bateria: En l'estat actual, el sistema necessita estar connectat a una font d'alimentació externa mitjançant cable. Per millorar-ne la portabilitat i usabilitat, especialment en entorns reals, seria recomanable incorporar una bateria recarregable que alimenti tant la placa ESP32 com els sensors.
- Entrenament del classificador directament des de l'aplicació mòbil: Actualment, l'entrenament del sistema es realitza exclusivament des de l'ordinador mitjançant terminal. Una millora significativa seria permetre a l'usuari registrar nous gestos i reentrenar el model K-NN des de l'aplicació mòbil, fent el sistema molt més autònom, interactiu i accessible per a qualsevol usuari sense coneixements tècnics.

Aquestes millores no només millorarien l'experiència de l'usuari final, sinó que permetrien adaptar el sistema a un ventall més ampli de necessitats i contextos d'ús, consolidant-lo com una eina útil per a la comunicació inclusiva.

## 8 Reflexió ètica i responsabilitat social

En aquest apartat es reflexiona sobre les principals dimensions ètiques i de responsabilitat social relacionades amb el projecte del guant traductor de llengua de signes.

### 8.1 Igualtat

El sistema desenvolupat està dissenyat per reconèixer gestos manuals associats a l'alfabet de la llengua de signes mitjançant sensors que capturen la flexió i orientació de la mà. Un aspecte clau és assegurar que aquest reconeixement funcioni de manera equitativa per a totes les persones, independentment del gènere, la mida de la mà o altres característiques físiques.

Per evitar possibles preferències de gènere o anatòmiques, s'ha incorporat un sistema de calibratge personalitzat que permet adaptar el dispositiu a les particularitats de cada usuari, ajustant els marges de detecció dels sensors. Aquest enfoc evita que el sistema depengui de valors estàndards que podrien beneficiar només un perfil concret, assegurant una experiència més justa i inclusiva.

A més, l'ús de l'algorisme K-Nearest Neighbors (K-NN), basat en la comparació amb mostres específiques de cada usuari, contribueix a reduir el risc de discriminació per característiques físiques, ja que cada usuari pot entrenar el seu propi conjunt de gestos.

### 8.2 Medi ambient

En la fase de disseny i selecció de components, s'ha tingut en compte la importància de la sostenibilitat ambiental. L'ús d'un microcontrolador com l'ESP32, conegut pel seu baix consum energètic, contribueix a minimitzar l'impacte ambiental durant l'ús del dispositiu.

A més, la tria de sensors senzills i de baix cost, així com l'ús de materials comuns per a la fabricació del guant, afavoreix la possibilitat de reciclatge o reutilització futura, reduint residus electrònics.

El sistema té la possibilitat de ser portable i eficient, el que implica una menor necessitat d'alimentacions externes constants i, per tant, un consum global més sostenible.

En futurs desenvolupaments, es pot considerar la incorporació de bateries recarregables d'alta eficiència i materials més respectuosos amb el medi ambient per reforçar aquest compromís.

### 8.3 Responsabilitat social

El projecte contribueix directament a la millora de la inclusió social i la comunicació entre persones amb discapacitat auditiva i la resta de la societat. Proporciona una eina que facilita la transmissió de missatges mitjançant la traducció automàtica de la llengua de signes, eliminant barreres lingüístiques i millorant l'accessibilitat.

Com a desenvolupador i futur professional, aquest projecte ha suposat també una presa de consciència sobre la importància de dissenyar solucions que no només siguin tècnicament vàlides sinó que tinguin un impacte positiu en la societat i que respectin la diversitat humana.

#### **8.4 Ètica**

Des del punt de vista ètic, el projecte s'ha desenvolupat respectant principis fonamentals com la transparència, la privacitat i el respecte pels drets dels usuaris.

El sistema no emmagatzema cap tipus de dades personals sensibles ni dades biomètriques més enllà dels gestos. A més, es reconeix la importància de garantir que la tecnologia no discrimini cap usuari.

Finalment, s'assumeix el compromís de mantenir una actitud crítica i oberta al diàleg, per continuar millorant el projecte, atenent les necessitats reals dels usuaris i promovent una tecnologia al servei de la inclusió social i el respecte mutu.

## 9 Conclusions

Aquest treball final de grau ha permès desenvolupar un sistema complet de reconeixement de gestos mitjançant un guant intel·ligent, combinant sensors de flexibilitat, orientació i una aplicació mòbil de suport. El resultat és un prototip funcional, portable i personalitzable, capaç de traduir gestos individuals en lletres mitjançant una classificació basada en mostres entrenades prèviament.

Des d'un punt de vista tècnic, s'ha assolit una integració efectiva de diferents tecnologies: sensors analògics, comunicació Bluetooth, microcontroladors, classificació per K-Nearest Neighbors i interfícies mòbils. Aquesta integració ha permès oferir un sistema modular i ampliable. Especialment rellevant ha estat la implementació d'un sistema de calibratge personalitzat, que permet adaptar el dispositiu a cada usuari, i la distinció entre gestos estàtics i dinàmics, que ha afegit robustesa al reconeixement.

Durant el desenvolupament s'han presentat diverses dificultats pràctiques: la inestabilitat de lectures, la fixació dels sensors, les confusions entre lletres similars o la gestió eficient dels recursos de la placa ESP32. Tots aquests reptes s'han abordat amb estratègies iteratives de prova i error, aportant valor a l'experiència com a enginyer.

Durant aquest procés he pogut aplicar coneixements adquirits en assignatures com Sistemes en Temps Real (STR), Aplicacions Mòbils i Encastades (AME), Anàlisi i Disseny d'Aplicacions (ADA) i Aplicacions i Serveis Mòbils (ASM). També he entès millor com la tecnologia pot adaptar-se a necessitats socials reals, i no només tècniques.

Finalment, aquest projecte ha servit per evidenciar el potencial transformador de la tecnologia quan s'aplica a problemes concrets i humans. L'objectiu no ha estat només construir un sistema que funcioni, sinó fer-ho amb una visió clara: contribuir a la comunicació i accessibilitat d'una manera senzilla i assequible. El resultat és un punt de partida sòlid per futures millores, que podrien incloure la detecció de paraules senceres, la incorporació de més idiomes de signes, o la connexió amb sistemes de reconeixement de veu o síntesi parlada.

## 10 Referències

- [1] Alfabet dactilològic de la llengua de signes catalana. <https://lsc.iec.cat/es/alfabet-dactilologico/>
- [2] SparkFun Thing Plus - ESP32 WROOM. <https://tienda.bricogeek.com/placas-adafruit-feather/1298-sparkfun-thing-plus-esp32-wroom.html> [consulta: 02/2025]
- [3] SparkFun Thing Plus - ESP32 WROOM – Esquema de pines [fitxer PDF]. [https://cdn.sparkfun.com/assets/learn\\_tutorials/8/5/2/ESP32ThingPlus\\_GraphicalDatasheet.pdf](https://cdn.sparkfun.com/assets/learn_tutorials/8/5/2/ESP32ThingPlus_GraphicalDatasheet.pdf) [consulta: 03/2025]
- [4] ESP32-WROOM datasheet [fitxer PDF]. [https://cdn.sparkfun.com/assets/learn\\_tutorials/8/5/2/esp32-wroom-32\\_datasheet\\_en.pdf](https://cdn.sparkfun.com/assets/learn_tutorials/8/5/2/esp32-wroom-32_datasheet_en.pdf) [consulta: 03/2025]
- [5] Sensor de flexibilitat. <https://amzn.eu/d/jdOheje> [consulta: 02/2025]
- [6] Naylamp Mechatronics. (s.d.). *Tutorial MPU6050: Acelerómetro y giroscopio*. [https://naylampmechatronics.com/blog/45\\_tutorial-mpu6050-acelerometro-y-giroscopio.html.\[12/03/2025\]](https://naylampmechatronics.com/blog/45_tutorial-mpu6050-acelerometro-y-giroscopio.html.[12/03/2025]) [consulta: 04/2025]
- [7] Sensor d'orientació MPU6050. <https://amzn.eu/d/bfT9n73> [consulta: 02/2025]
- [8] Fòrmules d'acceleració i velocitat angular. [https://naylampmechatronics.com/blog/45\\_tutorial-mpu6050-acelerometro-y-giroscopio.html.\[12/03/2025\]](https://naylampmechatronics.com/blog/45_tutorial-mpu6050-acelerometro-y-giroscopio.html.[12/03/2025]) [consulta: 04/2025]
- [9] Cover, T. M., & Hart, P. E. (1967). *Nearest neighbor pattern classification*. IEEE Transactions on Information Theory, 13(1), 21–27. <https://doi.org/10.1109/TIT.1967.1053964> [consulta: 05/2025]

## 11 Annex I. Posada en marxa i guia d'ús

Aquest annex proporciona una guia pràctica per posar en marxa el sistema desenvolupat, amb instruccions pas a pas per compilar, carregar i executar el programari del guant traductor de llengua de signes, així com per connectar i fer ús de l'aplicació mòbil associada.

### 11.1 Configuració de PlatformIO i llibreries

El desenvolupament del firmware per a la placa ESP32 s'ha dut a terme utilitzant Visual Studio Code amb l'extensió PlatformIO, que facilita la gestió del projecte, les llibreries i la càrrega del codi.

A continuació es mostra la configuració bàsica del fitxer platformio.ini:

```
[env:esp32thing_plus]
platform = espressif32
board = esp32thing_plus
framework = arduino
monitor_speed = 115200
lib_deps =
  mbed-syundo0730/I2Cdev@0.0.0+sha.3aa973ebe3e5
  jrowberg/I2Cdevlib-Core@^1.0.1
  electroniccats/MPU6050@^1.4.3
```

Figura 21. Configuració bàsica del fitxer platformio.ini

Aquestes llibreries són imprescindibles per a la lectura i gestió del sensor MPU6050, així com per a la comunicació I2C amb el microcontrolador.

### 11.2 Compilació i càrrega del firmware a la placa ESP32

Per compilar i carregar el firmware a la placa SparkFun Thing Plus ESP32 WROOM, cal seguir els passos següents dins de PlatformIO:

- 1- Connectar la placa via USB a l'ordinador.
- 2- Fer clic a "Build" per compilar el projecte.
- 3- Fer clic a "Upload" per carregar el firmware.
  - a. Durant el procés de càrrega, cal mantenir premut un botó físic de la placa ESP32 fins que comenci la transferència del codi.
- 4- Un cop carregat, fer clic a "Monitor" per iniciar la visualització en temps real del port sèrie i veure les dades i missatges de debug.

Com que el sistema encara no disposa de bateria, tant la placa com el dispositiu mòbil han d'estar connectats físicament a l'ordinador durant l'execució.

### 11.3 Execució de l'aplicació mòbil

L'aplicació Android desenvolupada per al projecte no ha estat publicada a cap botiga oficial, per tant, s'ha d'executar manualment des de l'entorn Android Studio:

- 1- Obrir el projecte de l'aplicació a Android Studio.
- 2- Connectar el telèfon Android per cable USB i activar el mode de depuració USB.
- 3- Fer clic "Run" per compilar i executar l'aplicació al dispositiu mòbil.

### 11.4 Funcionament de l'aplicació i calibratge

Un cop l'aplicació estigui executant-se en el mòbil i la placa estigui alimentada i operativa, es poden seguir aquests passos:

- 1- Prem el botó de "Connectar" a l'aplicació per iniciar la connexió Bluetooth amb l'ESP32.
- 2- Un cop connectat, es poden executar les rutines de calibratge:
  - a. Calibratge d'orientació: Es demana col·locar la mà en posició neutra.
  - b. Calibratge de flexibilitat: Es demana flexionar cada dit seguint les instruccions
- 3- Després del calibratge, el sistema comença a reconèixer els gestos realitzats i mostra la lletra detectada a la pantalla del mòbil.
- 4- L'usuari pot construir paraules mitjançant els gestos, fer ús de suggeriments de paraules i corregir errors amb el botó de "borrar".

### 11.5 Reinici del sistema

Si es vol reiniciar el calibratge, és necessari:

- 1- Parar l'execució del programa de la placa ESP32 i tornar a iniciar-lo.
- 2- Reiniciar l'aplicació mòbil, tancant-la completament i tornant-la a executar des d'Android Studio.