



**Universidad de Valladolid**



**ESCUELA DE INGENIERÍAS  
INDUSTRIALES**

**UNIVERSIDAD DE VALLADOLID**

**ESCUELA DE INGENIERIAS INDUSTRIALES**

**Grado en Ingeniería Electrónica Industrial y Automático**

# **SISTEMA AUTOMÁTICO DE DISPENSACIÓN DE MEDICAMENTOS**

**Autor:**

**Rodríguez Gobernado, Iván**

**Tutor:**

**González de la Fuente, José Manuel  
Tecnología Electrónica**

**Valladolid, julio de 2022.**



## Resumen

---

El deterioro cognitivo de la población conforme avanza en edad, con una población cada vez más envejecida, se ha convertido en un problema de orden internacional. Las soluciones comerciales son poco específicas y planteamientos más técnicos quedan fuera del alcance de los usuarios, viéndose relegados al papel en muchos casos. En el presente documento expondremos el desarrollo de un prototipo físico enfocado en el paciente, con un alto grado de personalización, modularización y accesibilidad. El enfoque de este trabajo es multidisciplinar, por lo que se desarrollará una solución integral capaz de responder a las necesidades de programación y mantenimiento de la persona responsable de cuidar al paciente, así como de los requerimientos de sencillez y comunicación de la persona dependiente.

## Palabras Clave

---

Alzheimer, Pastillero automático, Arduino, Aplicación Web, Automatización.

## Abstract

---

The cognitive deterioration of the population as it ages, with an increasingly older population, has become an international problem. Commercial solutions are not very specific and more technical approaches are out of reach for users, being relegated to paper in many cases. In this document we will present the development of a physical prototype focused on the patient, with a high degree of customization, modularization and accessibility. The focus of this work is multidisciplinary, so a comprehensive solution will be developed capable of responding to the programming and maintenance needs of the person responsible for caring for the patient, as well as the simplicity and communication requirements of the dependent person.

## Keywords

---

Alzheimer, Automatic pillbox, Arduino, Web Application, Automatization.



## Índice del contenido

Introducción y Objetivos.....	9
Contexto actual .....	9
Necesidades como objetivos.....	12
Estado del arte .....	13
Estudio de productos comerciales existentes .....	23
Pastilleros “analógicos” .....	23
Pastilleros digitales .....	24
Soluciones tecnológicas comerciales.....	25
Hardware del prototipo .....	29
Componentes del prototipo .....	29
Contenedor de medicamentos.....	29
Módulo RTC (Reloj de tiempo real) .....	31
Motor.....	32
Módulo de audio .....	33
Altavoces.....	34
Teclado matricial de membrana 4x4.....	35
Pantalla OLED.....	36
ESP01-S .....	37
Botón.....	38
Modelo cableado.....	39
Conexiones realizadas y pines en uso.....	40
Métodos de comunicación y control .....	41
Formas de alimentación .....	45
Modos de funcionamiento.....	46
Arduino Mega .....	47
Características.....	47
Justificación de la elección.....	48
Software creado .....	49
Base de datos del prototipo .....	49
Aplicación WEB.....	57
Lenguaje de programación, <i>framework</i> y módulos .....	57

Esquema de uso .....	58
Apartados de la página web .....	59
Aspectos del diseño.....	67
Sketch de Arduino .....	69
Bibliotecas empleadas .....	69
Esquema de uso .....	70
Aspectos de la construcción del prototipo.....	72
Conclusiones.....	75
Visión global del proyecto .....	75
Consecución de objetivos .....	75
Aspectos de mejora .....	77
Bibliografía .....	81

## Ilustraciones

---

Ilustración 1 - Fragmento de la Infografía sobre la Demencia (World Health Organization, 2020).....	9
Ilustración 2 - Fragmento de Infografía sobre el Plan de actuación contra la demencia de la OMS (World Health Organization, 2020).....	11
Ilustración 3 - Detalle del prototipo de pastillero.....	13
Ilustración 4 - Detalle del pastillero modificado .....	14
Ilustración 5 - Detalle del prototipo de caja-pastillero.....	15
Ilustración 6 - Distribución de los componentes del pastillero.....	16
Ilustración 7 - Detalle de la aplicación desarrollada .....	16
Ilustración 8 - Dispensador de medicamentos .....	17
Ilustración 9 - Pastillero rotacional para pacientes con trastorno bipolar.....	18
Ilustración 10 - Esquema de interconexión del pastillero con comunicación bidireccional .....	19
Ilustración 11 - Esquema del pastillero automático.....	20
Ilustración 12 - Pastillero asistido por móvil .....	21
Ilustración 13 - Detalles de la aplicación móvil .....	21
Ilustración 14 - Concepto de pastillero.....	22
Ilustración 15- Representación por bloques del esquema de conexiones.....	39
Ilustración 16 - Arduino Mega .....	47
Ilustración 17 - Esquema de uso de la aplicación.....	58
Ilustración 18 - Esquema de uso del prototipo por modos.....	70
Ilustración 19 - Esquema del cambio en el modo de funcionamiento del prototipo.....	71

## Gráficas

---

Gráfica 1 - Prevalencia de la Demencia por tramos de edad. ....	10
Gráfica 2 - Tensión continua equivalente por alteración de la activación a lo largo de un ciclo .....	43
Gráfica 3 - Modulación de una señal mediante el uso de PWM.....	44

## Figuras

---

Figura 1 - Pastillero comercial.....	23
Figura 2 - Pastillero electrónico.....	24
Figura 3 - Envasadora semi-automática de medicamentos .....	26
Figura 4 - Contenedor de bolsas .....	26
Figura 5 - Envasadora automática de medicamentos.....	27
Figura 6 - Máquina de control y recortadora.....	27
Figura 7 - Dispensador de bolsas contenedoras .....	28

Figura 8 - Desglose de las piezas que componen el dispensador .....	29
Figura 9 - Mapa de pines del Arduino Mega .....	48
Figura 10 - Esquema de la estructura de la base de datos .....	49
Figura 11 - Esquema de tratamiento de datos.....	50
Figura 12 - Prototipo elaborado como TFG.....	72
Figura 13 - Alimentación de la placa de prototipado .....	72
Figura 14 - Parte superior del pastillero .....	73
Figura 15 - Dispensadores acoplados en el pastillero.....	73
Figura 16 - Detalles de las placas de prototipado soldadas con los componentes modulares .....	74
Figura 17 - Capturas de la aplicación de escritorio que se empezó a elaborar y se migró.....	79

## Tablas

---

Tabla 1 - Resumen de las conexiones series que presentan distintos modelos de Arduino.....	42
Tabla 2 - Tabla AppUser de la base de datos .....	51
Tabla 3 - Tabla Administration de la base de datos .....	52
Tabla 4 - Tabla Message de la base de datos .....	52
Tabla 5 - Tabla Medicine de la base de datos.....	53
Tabla 6 - Tabla Patient de la base de datos .....	54
Tabla 7 - Tabla Intake de la base de datos.....	55
Tabla 8 - Tabla Posology de la base de datos .....	56

## Fragmentos de la aplicación Web

---

WebApp 1 - Pestaña de acceso .....	59
WebApp 2 - Página principal.....	59
WebApp 3 - Apartado de pacientes .....	60
WebApp 4 - Formulario para el registro de un nuevo paciente.....	61
WebApp 5 - Apartado Medicamentos.....	62
WebApp 6 - Formulario para el registro de un medicamento.....	63
WebApp 7 - Apartado de Pastilleros (Selector).....	63
WebApp 8 - Apartado de Pastilleros (personal).....	64
WebApp 9 - Formulario para el registro de tomas.....	65
WebApp 10 - Gestor de Momentos de la ingesta.....	65
WebApp 11 - Formulario para el registro de momentos de ingesta .....	66
WebApp 12 - Apartado de mensajes .....	66
WebApp 13 - Página inicial de presentación .....	67
WebApp 14 - Paleta de colores empleada en la aplicación .....	68



## Introducción y Objetivos

### Contexto actual

Con el aumento de la esperanza de vida y el envejecimiento de la ciudadanía europea, así como la creciente sensibilidad hacia la calidad tanto física como psíquica de las personas en aras de granjearles una vida digna, el problema de las demencias, del que ya se habla como la **gran epidemia silenciosa del siglo XXI**, se ha convertido en un importante reto que pone en jaque el buen funcionamiento del sistema social y sanitario.

Según la definición de (World Health Organization, 2020):

*La demencia es un síndrome –generalmente de naturaleza crónica o progresiva– caracterizado por el deterioro de la función cognitiva (es decir, la capacidad para procesar el pensamiento) más allá de lo que podría considerarse una consecuencia del envejecimiento normal. La demencia afecta a la memoria, el pensamiento, la orientación, la comprensión, el cálculo, la capacidad de aprendizaje, el lenguaje y el juicio. La conciencia no se ve afectada. [...]*

*La demencia es una de las principales causas de discapacidad y dependencia entre las personas mayores en todo el mundo. [...]*

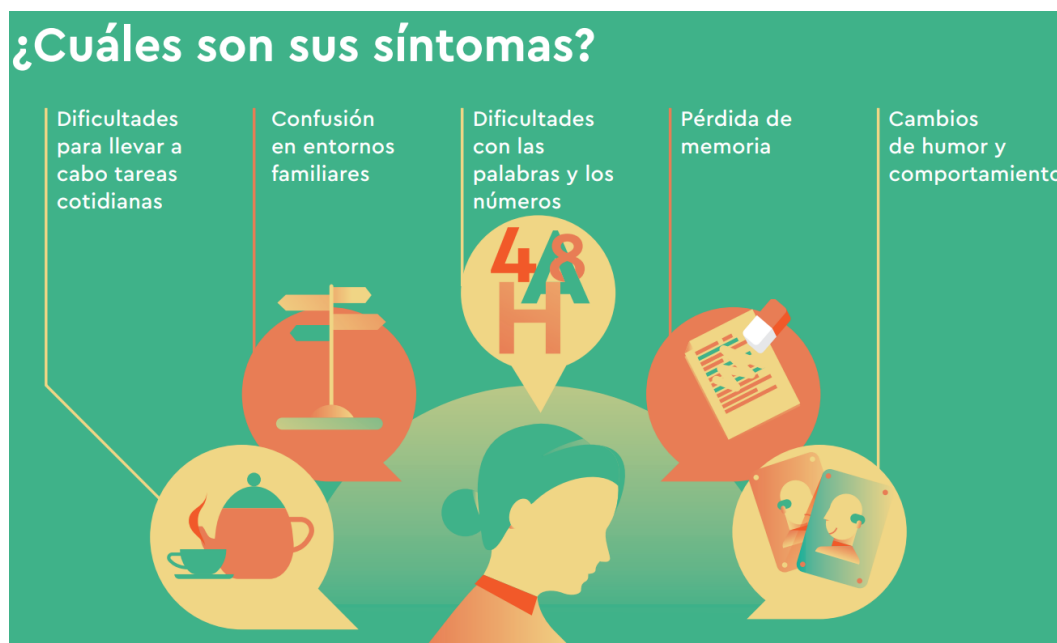


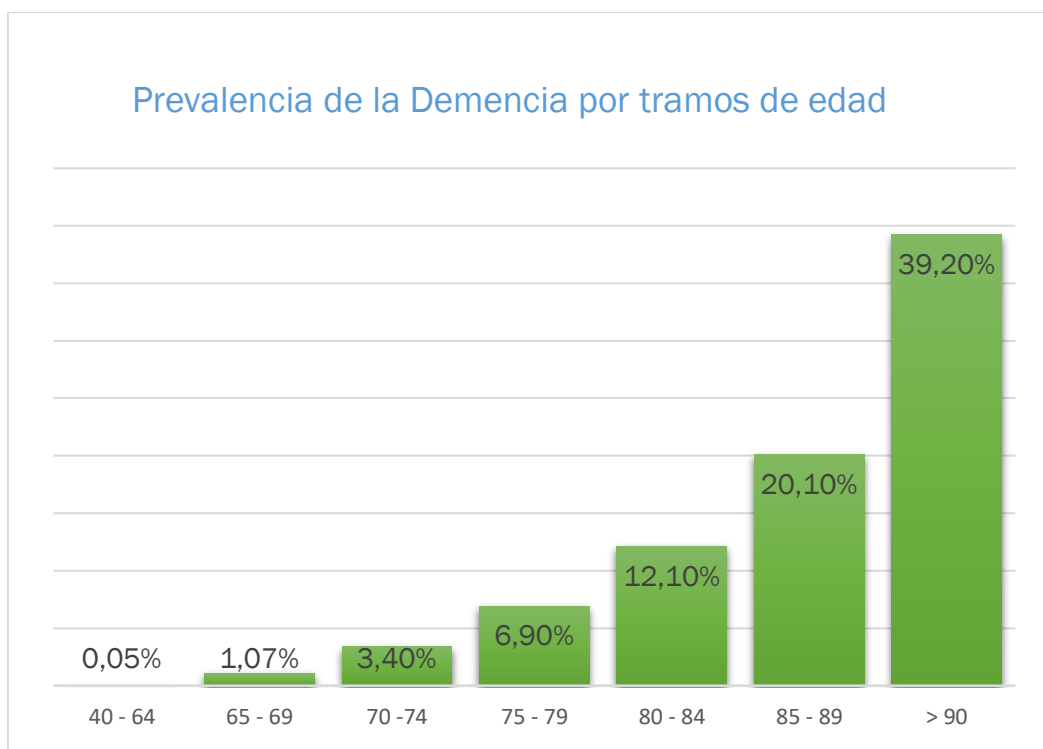
Ilustración 1 – Fragmento de la Infografía sobre la Demencia (World Health Organization, 2020).

Entre un 5% y un 8% de las personas de 60 años en adelante padece demencia, registrándose 10 millones de casos nuevos cada año, lo que supone el descubrimiento de un nuevo caso cada 3 segundos. Para la fecha de publicación del artículo anteriormente citado, la acumulación era de 50 millones de afectados en todo el mundo, siendo las previsiones de 82 millones de casos en 2030 y llegando a alcanzar los 152 millones en 2050, en pocas palabras, nos encontramos ante una problemática creciente.

Según (Ministerio de Sanidad, Consumo y Bienestar Social, 2019):

*Tomando los datos poblacionales del INE y aplicando esas cifras de prevalencia, el número de personas afectadas en España supera las 700.000 personas entre los mayores de 40 años. En 2050 el número de enfermos se habrá duplicado y se acercará a los dos millones de personas.*

En la siguiente gráfica se resumen los datos relativos a la prevalencia de la demencia por tramos proporcionada por el Gobierno de España y expuestos en el documento: (Ministerio de Sanidad, Consumo y Bienestar Social, 2019).



Gráfica 1 - Prevalencia de la Demencia por tramos de edad.

## Visión

Lograr un mundo en el que se pueda prevenir la demencia y las personas con demencia y sus cuidadores vivan bien y reciban la atención y el apoyo que necesitan para realizar su potencial con dignidad, respeto, autonomía e igualdad.

## Objetivo

Mejorar la vida de las personas con demencia, sus cuidadores y sus familiares, y reducir el impacto de la demencia sobre ellos y sobre las comunidades y los países.

*Ilustración 2 - Fragmento de Infografía sobre el Plan de actuación contra la demencia de la OMS (World Health Organization, 2020).*

La OMS aprobó el 29 de mayo de 2017 el “Plan de acción mundial sobre la respuesta de salud pública a la demencia de la OMS”, para inspirar a los Estados en el abordaje del problema. El gobierno de España, en consonancia con estas directrices, redactó el Plan Integral de Alzheimer y otras Demencias con marco de actuación 2019-2023 (Ministerio de Sanidad, Consumo y Bienestar Social, 2019), en el cual se establecen “seis factores de dimensión, alcance e impacto en demencias”:

*Factor 1 – La elevada prevalencia de las demencias.*

*Factor 2 – Las demencias son una de las principales causas de discapacidad y dependencia y suponen graves y progresivas limitaciones para la autonomía en las actividades básicas e instrumentales de la vida diaria.*

*Factor 3 – El Alzheimer [demencia] es devastador y genera graves impactos, no sólo para quienes padecen la enfermedad, sino también para sus familiares cuidadores.*

*Factor 4 – El Alzheimer y otras demencias plantean importantes retos para la sostenibilidad de los sistemas de protección social.*

*Factor 5 – Contexto de déficit y fragmentación de la respuesta social y sanitaria a las necesidades de prevención, tratamiento y atención.*

*Factor 6 – Contexto generalizado de falta de conocimiento, conciencia y comprensión sobre el alzhéimer y de déficit de participación de la propia persona.*

## Necesidades como objetivos

Como los datos anteriormente expuestos apuntan, la demencia es un problema creciente con un impacto internacional y que supone un gran reto debido a su repercusión multinivel.

Este síndrome incapacita en muchos casos a las personas que la padecen privándoles de la autonomía a la hora de realizar tareas diarias o, simplemente, imposibilitando el disfrute de su vida de forma independiente. Un acto tan común en personas de edad avanzada como la toma de los medicamentos prescritos por el personal sanitario se puede ver obstaculizada tanto por el olvido de la actividad, como por la confusión en el reconocimiento de los diferentes fármacos, la desorientación de la persona, etc. Este hecho es potencialmente peligroso cuando entran en juego diferentes tipos de medicamentos cuya ingesta regular puede acarrear efectos nocivos en la persona, o la propia sobre medicación.

Este *Sistema Automático de Dispensación de Medicamentos* pretende ayudar a una persona real para preservar su independencia, el trabajo se plantea como el desarrollo de un prototipo capaz de implementar diferentes funcionalidades que cubrirán las necesidades específicas básicas adaptadas a las circunstancias de la persona de modo que, en una revisión posterior del modelo, se pueda obtener un dispositivo final adaptado a la evolución de la relación entre el interfaz y la persona.

Los objetivos, por tanto, se concretan en necesidades del paciente al que va dirigido la aplicación del prototipo:

- Interacción con el usuario a nivel Hardware con el dispositivo simple y minimalista.
- Interacción con el usuario a nivel software simple pero completa.
- Restricción de acceso a los medicamentos.
- Avisos suficientemente evidentes.
- Comunicación visual básica mediante Hardware del estado del dispositivo.
- Diseño modular para maximizar la adaptabilidad a necesidades no contempladas.
- Capacidad de recolección de datos y su transmisión a una persona responsable.
- Capacidad de comunicación del proceso de una forma profesional de modo que la persona sea conocedora de la información relativa a la toma del medicamento.

## Estado del arte

En este apartado vamos a exponer el resultado de la búsqueda de diferentes trabajos que presentan soluciones ante este problema planteado. Todos los trabajos citados han sido buscados en el repositorio IEE Xplore.

1. (Parra, Valdez, Guevara, Cedillo, & Ortiz-Segarra, 2017).

Se trata de un primer acercamiento al problema mediante el uso tecnología de asistencia. El prototipo incluye un sistema de alarmas y un cierre automatizado, el controlador sobre el que se monta es un Arduino Mega 2560 y cuenta con un sistema de interfaz de usuario interactivo.



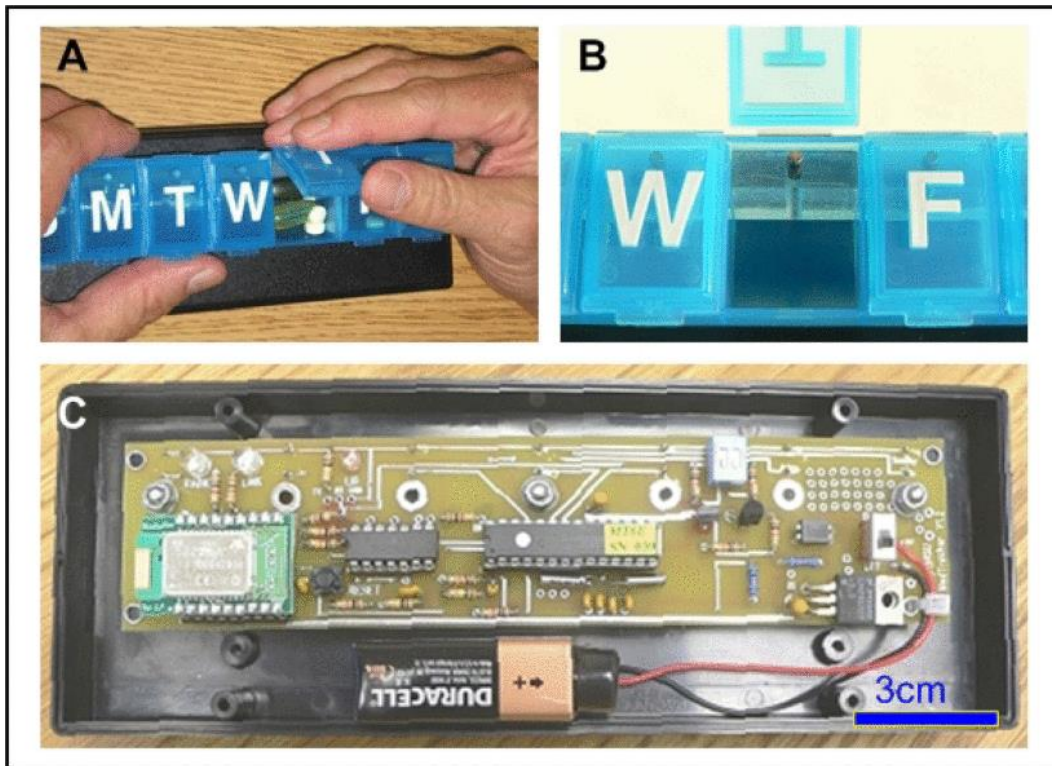
*Ilustración 3 - Detalle del prototipo de pastillero*

Como se puede apreciar en la ilustración, el dispositivo cuenta con una pantalla táctil de cinco pulgadas, dos compuertas diferenciadas e independientes y sensores infrarrojos para la detección de la manipulación de medicamentos.

La capacidad de comunicación con el cuidador, así como la detección de la manipulación de los medicamentos son puntos muy positivos para destacar, por el contrario, la forma en la que se abren los compartimentos deja a la vista demasiados casilleros a la vez, lo que puede confundir al paciente.

2. (Hayes, Hunt, Adami, & Kaye, 2006).

Esta aproximación toma un concepto existente y archiconocido de pastillero y eleva sus características en el ámbito del monitoreo, las celdas son modificadas de modo que un interruptor, que se activa cuando estas sean abiertas, registra el evento.



*Ilustración 4 - Detalle del pastillero modificado*

El concepto principal de este proyecto reside en la adaptación de un elemento familiar para el paciente, lo cual es un concepto de partida muy interesante y un punto fuerte del proyecto, no obstante, deja a un lado el papel de gestión de medicamentos y tanto la seguridad como las indicaciones queda relegadas a un segundo plano.

En la Ilustración 4, podemos apreciar la placa de circuitos diseñada, comprobando que se ha seguido un muy buen proceso de síntesis de síntesis consiguiendo una gran reducción del espacio necesario, por tanto, la portabilidad es un punto muy a favor de este diseño.

3. (Jabeena, Sahu, Roy, & Basha, 2018).

Este prototipo se nos presenta como un concepto de bajo coste, aprovechando materiales reciclados y con una filosofía positivista en cuanto a su planteamiento.



*Ilustración 5 - Detalle del prototipo de caja-pastillero*

Este proyecto si bien es interesante por lo reducido que puede resultar el coste y lo accesible de su construcción, no representa una solución que se adecue a los parámetros de seguridad que cabría esperar, la ubicación del sensor de infrarrojos es ciertamente optimista en cuanto a las detecciones y por dentro cuenta con unos compartimentos muy grandes pensados para el almacenaje masivo de los medicamentos, lo cual no suele representar una solución que se adapte a las necesidades de una persona mayor.

Uno de los puntos a favor de este proyecto es la capacidad de comunicación que presenta y en la descripción de este se hace referencia a aplicaciones más relacionadas con la de servir de apoyo a personas sin hábitos en la toma de medicamentos, aplicados a estos campos, la solución si parece adecuarse.

4. (Rajan, y otros, 2021).

Esta solución cuenta con un alto grado de sensorización, si bien el planteamiento adolece en la parte de la dispensación, almacenaje y seguridad de los medicamentos, cuenta con su propia aplicación, Ilustración 7, y comunicación con la misma, generando un producto muy completo.

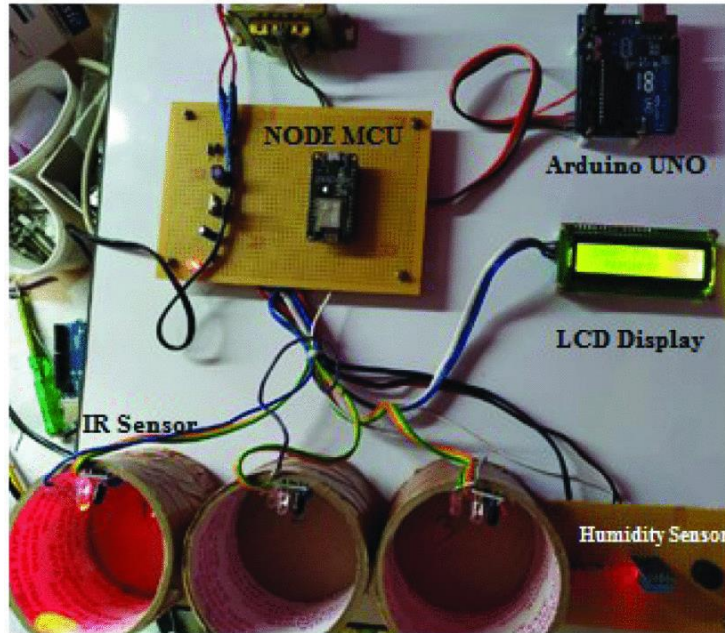


Ilustración 6 - Distribución de los componentes del pastillero

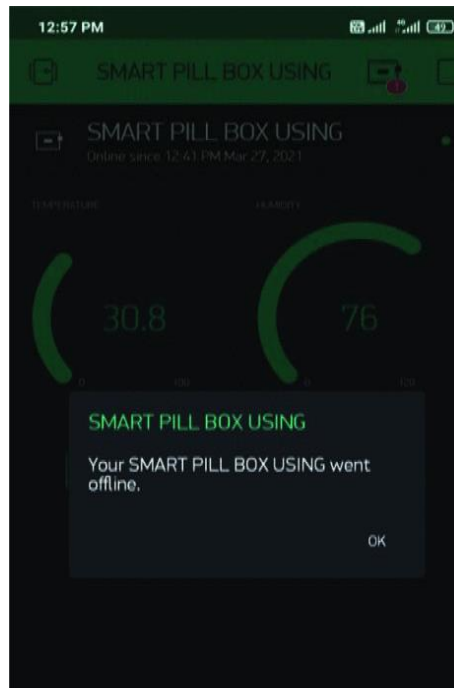
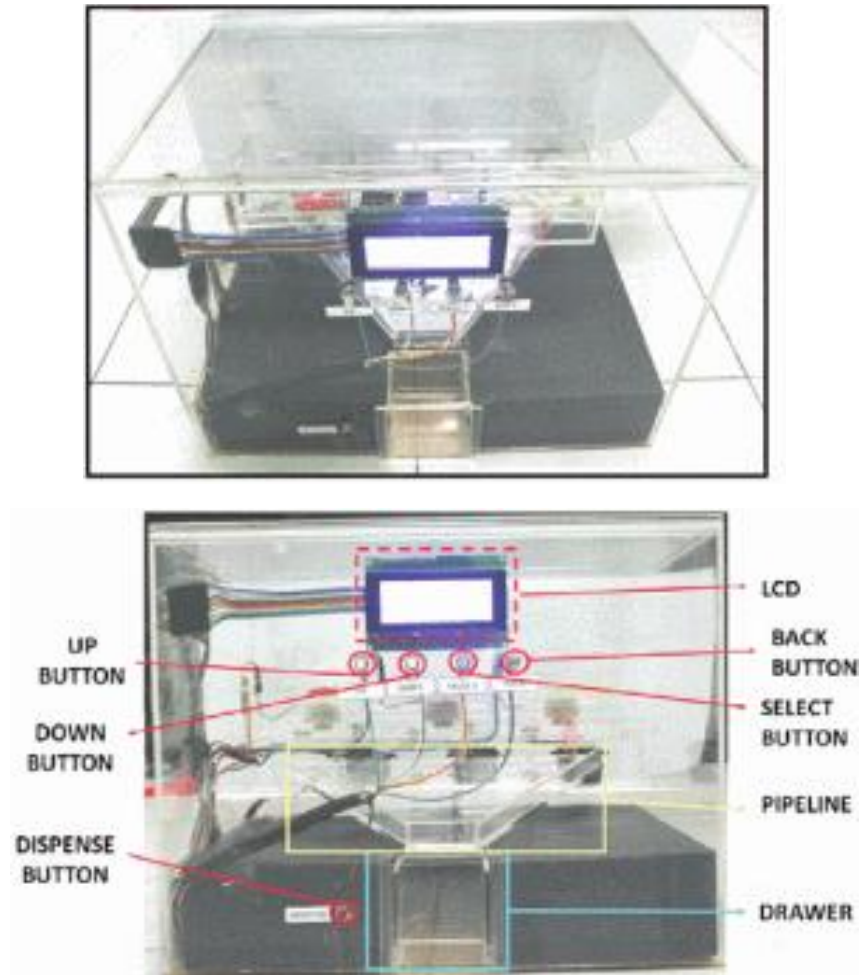


Ilustración 7 - Detalle de la aplicación desarrollada



5. (Othman & Ek, 2016)

Este prototipo es uno de los más completos y resolutivos en su ejecución. No solo cuenta con el dispensador de medicamentos, también es capaz de enviar mensajes que alertan al cuidador de los eventos que se producen.



*Ilustración 8 - Dispensador de medicamentos*

El funcionamiento del dispensador es muy curioso, la dispensación se produce haciendo vibrar el habitáculo de las pastillas hasta que el sensor infrarrojo detecta que una ha caído, lo cual supone un buen método para almacenar los medicamentos con una recarga sencilla y una comprobación eficaz.

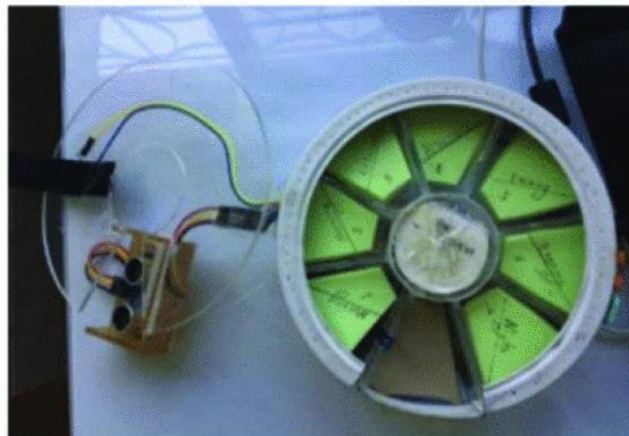
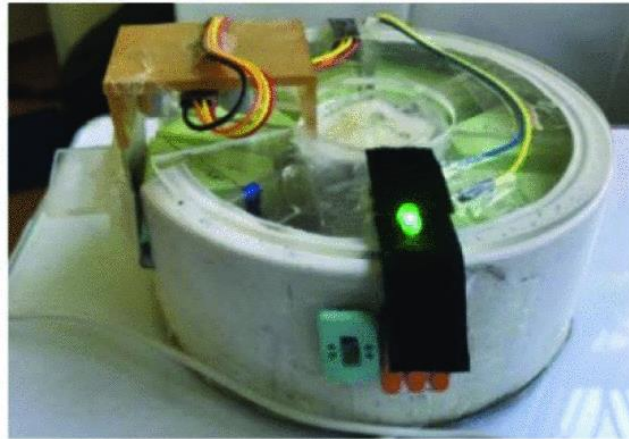
El tamaño podría ser uno de los puntos débiles del planteamiento, pues es necesario por las técnicas de dispensación empleadas que este sea un gran bloque, lo cual imposibilita la adaptación a modelos más reducidos.

6. (Kanhasinwattana, Yawila, Tithada, & Kamyod, 2020).

Si bien el planteamiento está más centrado para pacientes con trastorno bipolar, por las características de este, se ha entendido que puede ser objeto de análisis.

El proyecto cuenta con su propia aplicación móvil con la que el prototipo establece comunicación. Funciona empleando un servomotor que gira en torno hasta el casillero necesario, un sensor de ultrasonidos es el encargado de detectar si se ha producido o no la toma de medicamentos.

Nos encontramos, por tanto, ante una solución muy completa, escalable y que se adapta bien al entorno comercial actual.



*Ilustración 9 - Pastillero rotacional para pacientes con trastorno bipolar*

7. (Tsai, Tseng, Wang, & Juang, 2017).

Se trata de una de las soluciones más profesionales que se han encontrado en la búsqueda de casos desarrollados. No solo cuenta con la capacidad de enviar mensajes de manera bidireccional, como destacan los autores, también es una solución compacta, robusta y bien estructurada, con una gran capacidad de adaptación a los diferentes eventos que puedan ocurrir.

Otro punto muy destacable es la gran labor llevada a cabo a la hora de pulir la interfaz del pastillero, reducida a la mínima expresión y con gran capacidad comunicativa.



Ilustración 10 - Esquema de interconexión del pastillero con comunicación bidireccional

8. (Rosdi, y otros, 2021).

Este pastillero toma un concepto existente de forma comercial, muy empleado, muy simple y reconocible, y lo transforma automatizando el proceso de elección del casillero. Cuenta con una interfaz reducida y minimalista, un servomotor que modula el giro, un sistema de alertas visual y acústico de impacto y capacidad de comunicación con una aplicación propia.

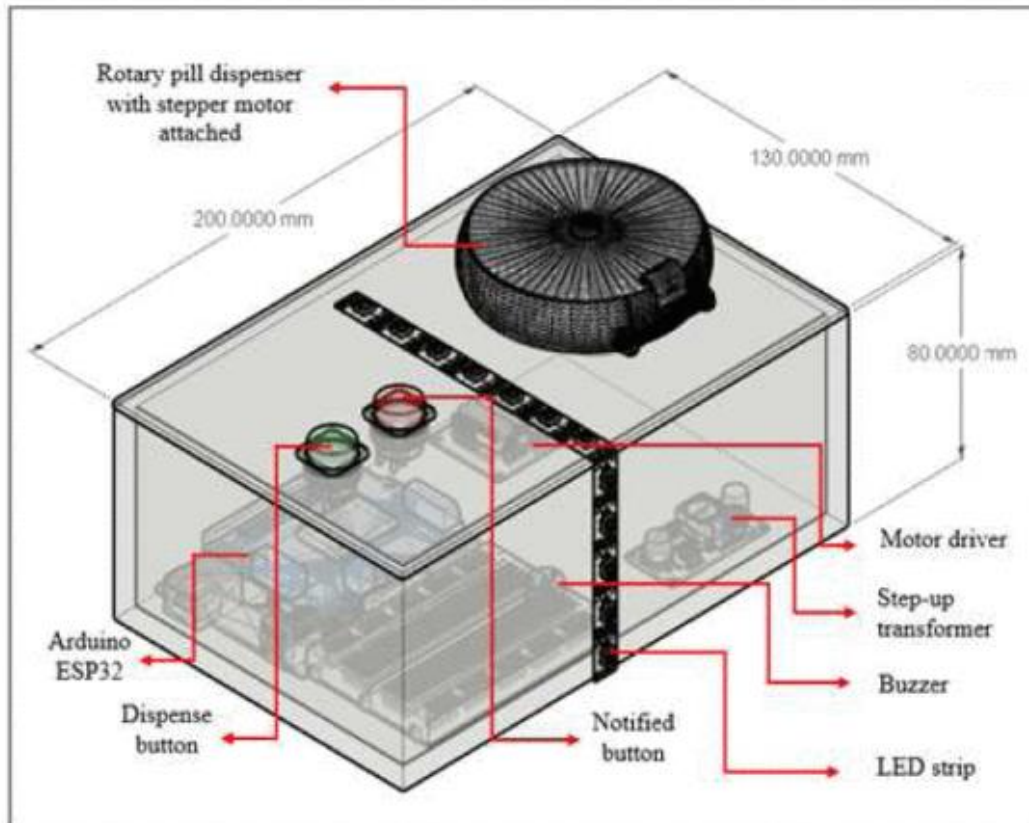
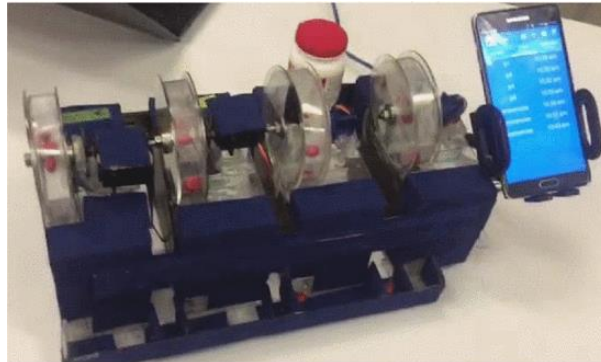


Ilustración 11 - Esquema del pastillero automático

9. (Antoun, y otros, 2018).

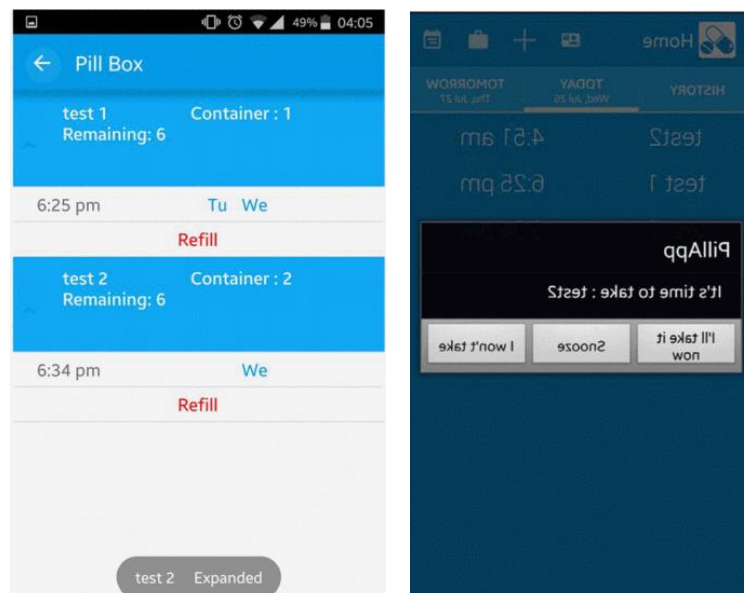
Este proyecto aúna en un mismo punto su punto más relevante y lo que podría suponer la mayor falla en la accesibilidad de este, emplea un smartphone como base operativa del mismo, más concretamente, en la aplicación desarrollada.



*Ilustración 12 - Pastillero asistido por móvil*

Como se puede apreciar, el sistema tiene una gran capacidad de carga y está pensado para su uso estático. Una posible revisión podría adaptar su empleo de modo que el control se desligue de la aplicación móvil pues para personas mayores estos dispositivos son difíciles de usar.

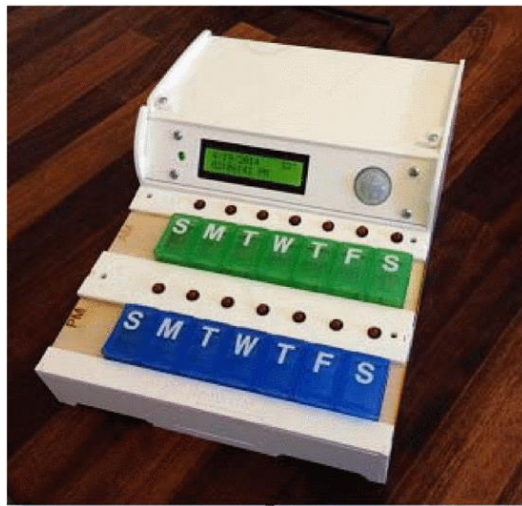
Emplea servomotores para el giro de los contenedores, su controlador se basa en un Arduino, tiene conexión directa con una base de datos en la nube por medio de la aplicación telefónica y la comunicación con el prototipo se realiza mediante bluetooth.



*Ilustración 13 - Detalles de la aplicación móvil*

10. (Al-Shammary, Mousa, & Esmaeili, 2018).

El concepto de este pastillero es similar al comentado en (Hayes, Hunt, Adami, & Kaye, 2006), sin embargo, se incluye debido a su capacidad de adaptación del concepto a un diseño propio. La inclusión de una sensorización por medio de imanes y de un módulo x-bee reemplazan a las elecciones del anteriormente citado proyecto de forma orgánica. Por otro lado, este sistema incluye, además, indicaciones sonoras y visuales, completando las funcionalidades de que carecía el anterior.



*Ilustración 14 - Concepto de pastillero*

Podemos concluir de esta pequeña búsqueda de soluciones que hemos llevado a cabo, que existen múltiples opciones válidas para atajar los problemas que en el presente documento venimos refiriendo, si bien muchas de ellas se olvidan de tres elementos básicos a la hora de tratar con un paciente con Alzheimer, la seguridad que ha de transmitir por medio del bloqueo al acceso de los medicamentos, la capacidad de facilitar el proceso al cuidador en el plano administrativo y la necesidad de una interfaz extremadamente sencilla.

Por todo esto es que se ha decidido elaborar una solución propia para un paciente real, dotándola, además, de algo que se hace fundamental si nos paramos a reflexionar sobre el tema. Si para un paciente concreto se necesitan una serie de especificaciones que un buen número de proyectos no son capaces de satisfacer al cien por cien, la consecución de un prototipo que sea lo suficientemente modular y adaptable como para adecuarse a las necesidades del paciente, nos brindará una base de trabajo muy importante no solo para futuras revisiones, sino para personas que, con necesidades muy específicas, acudan a este prototipo para modificarlo a conveniencia.

## Estudio de productos comerciales existentes

Si realizamos una búsqueda de las diferentes soluciones que están a la venta, podemos fijarnos en dos tipos de enfoques diferentes a la hora de plantear la finalidad del producto.

### Pastilleros “analógicos”

Por un lado, tenemos pastilleros orientados a la organización eficiente de los medicamentos. Estos pastilleros de tipo casillero, cuentan con unas celdillas marcadas siguiendo un patrón temporal, de modo, dividen las distintas tomas atendiendo a un criterio generalista en las tres comidas principales extendiendo a una cuarta toma para la hora de dormir o el desayuno en muchos casos y que repiten el patrón para abarcar periodos de tiempo diferentes, dependiendo del modelo, pudiendo encontrarnos con pastilleros para el día, semanales, etc.



*Figura 1 - Pastillero comercial*

Estos pastilleros son los más económicos, compactos en su diseño y versátiles, puesto que la gran mayoría de pastilleros que se extienden en grandes periodos (suele ser el caso de los semanales), tienen la opción de extraer una fila o columna, según el caso, relativa a la toma del día.

El principal y más evidente problema de estos pastilleros radica en la simplicidad de su diseño, pues no permite incorporar alarmas ni otro tipo de personalización, más allá de meras decoraciones que ligeramente tienen mayor funcionalidad que la estética.

## Pastilleros digitales

Por otro lado, existen pastilleros programables. Estos permiten establecer alarmas e incluso automatizan el proceso, ya sea dispensando el medicamento que en el momento corresponde o, como en el caso de la imagen, un bombo rotativo pone el casillero correspondiente en posición, es decir, automatizan el proceso de selección de la celdilla.



*Figura 2 - Pastillero electrónico*

Este tipo de pastilleros, aun siendo asequibles en muchos casos, elevan el precio del producto en su inclusión de componentes electrónicos, son menos versátiles pues la no modularización de los mismos nos deja con un producto o muy grande en dimensiones y con una elevada capacidad, o muy reducido, más práctico, pero más tedioso de recargar.

Los principales problemas son, la dificultad de uso, que en muchos casos “marginan” a personas mayores, lo aparatoso de los modelos muy voluminosos, lo poco accesible que resulta para personas con problemas de psicomotricidad en las manos el manejo de cierres de seguridad y el grado de personalización no solo a nivel de las alarmas, sino, también, de la adaptación a las necesidades del usuario. De aquí la importancia de un planteamiento más modular en cuanto a la construcción del prototipo que en este documento estamos tratando, simplificando el uso del dispositivo e incluyendo gran capacidad de interacción con el usuario.



## Soluciones tecnológicas comerciales

Podemos comprobar que estas soluciones más tradicionales o mejoran la organización en los modelos más básicos o, yéndonos a modelos más caros y completos, incluyen alertas, pero se quedan cortos en capacidad de adaptabilidad por ser, como es obvio para su planteamiento demasiado generalistas.

Estas fallas que hemos remarcado en apartados anteriores pueden no representar un gran inconveniente para el público general, teniendo en cuenta, además, que la edad no es un factor limitante a la hora de empezar a tomar algún tipo de medicamento, sin embargo, para personas con Alzheimer estas soluciones son, según el grado de afectación de la enfermedad, o incompletas o totalmente inadecuadas.

Este hecho es notorio no solo para personas que cuenten en su familia con personas mayores con algún tipo de degeneración cognitiva por la edad, si no para todos aquellos que o bien conozcan algún caso de primera mano o se acercan al tema pues la empatía nos permite vislumbrar sin tener que sumergirnos en el grueso de casos e información médica relativa a estos temas, la disminución en la calidad de vida que supone el padecimiento de esta dolencia. Como no podía ser de otro modo, el problema no pasado desapercibido para las compañías que tienen alguna línea de productos en esta dirección o que quieren introducirse en el mercado de manera efectiva y efectista y ya hay soluciones personalizadas muy interesantes.

En este apartado, trataremos uno de los múltiples casos que pueden encontrarse tras una rápida búsqueda en internet de mano de prestigiosas e importantes compañías como pueden ser Samsung o Siemens entre otras. El caso que expondremos es notorio por implementar una solución desde la raíz del problema, planteando un reacondicionamiento del empaquetado de los medicamentos lo que, como podemos imaginar, repercute también, de forma positiva en el medioambiente derivado de la simplificación del empaquetado de medicamentos.

Desde la página web de la empresa (Ti-Medi, s.f.), podemos observar el conjunto de soluciones planteado de forma integral que se ofrece a farmacéuticas, residencias y particulares.



*Figura 3 - Envasadora semiautomática de medicamentos*

El enfoque puede ser farmacéutico, con productos capaces de extraer del paquete los medicamento y de emplazarlos de manera semiautomática. La idea es pasar del archiconocido blíster a un formato que no suponga al paciente conservar varias cajas en su casa, aprenderse las diferentes formas y colores de cada medicamento que ha de tomar y acordarse de qué caja contiene el medicamento de cada toma.



*Figura 4 - Contenedor de bolsas*

El trabajo se enfoca en el empaquetamiento de los medicamentos en bolsas, en la Figura 4 se puede ver un contenedor que sirve para su empaquetamiento y puede ser conectado a otros productos para la inspección o el transporte.



*Figura 5 - Envasadora automática de medicamentos*

No obstante, la compañía tiene enfoques mucho más automatizados, como el que podemos ver en la Figura 5. Un armario que contine gran variedad de medicamentos, cuando se transmite la orden de empaquetado, estos se almacenan en las bolsas ya mencionadas.



*Figura 6 - Máquina de control y recortadora*

Implementado, incluso, su propio sistema de control del proceso. El reconocimiento es visual, las bolsas que contienen los medicamentos pasan por la máquina que primero detecta el contenido que debería estar presente en dicha bolsa y después se cerciora de que así sea. La empresa también cuenta con su propio sistema de cortado de bobinas, lo que les permite separar las bolsas que presenten un contenido defectuoso.



*Figura 7 - Dispensador de bolsas contenedoras*

Como se puede apreciar uno de los principales problemas de este método de dispensación es la falta de control que el paciente con demencia va a tener sobre cuándo ha de tomar la dosis y si ya ha tomado la correspondiente. No olvidemos que la desorientación es uno de los principales síntomas de la demencia y, si lleva a la sobre medicación, una potencial causa de afecciones graves e incluso la muerte.

La solución que la empresa plantea es muy interesante, permite una gran capacidad de respuesta y personalización en el trato farmacológico, automatizando un tedioso proceso, personalizando el producto y reduciendo los empaquetados necesarios. Una solución tecnológica y técnica que continúa y mejora los ya incipientes procesos de automatización de las farmacias, libera de carga de trabajo a los farmacéuticos y les permite tener un trato más cercano y desarrollarse más como asesores del paciente que como comerciales. No obstante, los problemas son los que ya hemos comentado, si bien la línea de productos es muy aplicable al ámbito farmacológico e incluso geriátrico, decaen a la hora de tratar a particulares con características tan especiales como las personas que sufren Alzheimer.

Sería interesante, plantear un complemento a este producto en forma de un pastillero con las características que el que en este trabajo se expone si soluciones como las que la empresa ofrece adquieren fuerza en forma de presencia en el sector y una cantidad relevante de pacientes se interesan por el producto de forma que lo incorporen a su rutina.

## Hardware del prototipo

Para la parte operativa del prototipo se han elegido unos módulos compatibles con Arduino para atajar las necesidades más generales y que tienen solución comercial. Para la parte móvil del dispensador, que hará las veces de contenedor de los medicamentos hasta el momento en el que se hayan de dispensar, por lo específico del prototipo y en aras de cubrir las necesidades del proyecto, se ha diseñado un modelo 3D para cubrir el propósito.

### Componentes del prototipo

#### Contenedor de medicamentos

Empezaremos hablando del modelo 3D que es la pieza sobre la que tenemos más control por estar fabricada ad hoc para el prototipo. Necesitábamos satisfacer las características que al inicio de este documento hemos descrito. El modelo final al que se ha llegado tras un proceso de diseño y un posterior ajuste por prueba y error está basado en el uso de un servomotor, con el procesado de las partes impresas y la aplicación de agentes lubricantes, en este caso, vaselina neutra, se consigue la reducción de la fricción hasta el punto de que un servomotor pequeño podría mover el mecanismo, no obstante, emplearemos un micro servo 9g por ser de los más comunes, siendo fácilmente adquirible en varias páginas de compra online, y económicos a la hora de plantear un proyecto basado en controladores tipo Arduino.



*Figura 8 - Desglose de las piezas que componen el dispensador*

Se ha construido una carcasa, compuesta por una cavidad en la que será introducido el motor y asegurado mediante el empleo de un tornillo y un pilar que se ha vaciado para que pueda contener el medicamento que se desea dispensar con posterioridad esta es la pieza con mayor cantidad de material del dispensador, dejando la tapa, que es más liviana, como un elemento móvil, reduciendo así la carga que ha de vencer el motor para poder hacer efectiva la entrega de la dosis pertinente.

La transmisión se va a realizar mediante engranajes, como el movimiento requerido se puede resumir como un giro de 90 grados y la capacidad de giro del servo es de 360, se ha diseñado una reducción 2:1 de modo que con 180 grados de rotación del motor podamos abrir el compartimento, así si existe alguna falla en la capacidad de giro del motor, este puede ser ajustado truncando su cero y compensando el fallo para que no mengue la capacidad de rotación de este. Uno de los engranajes se acopla al servomotor y, aprovechando los acoples que los servomotores de este tipo incluyen, emplearemos uno de los disponibles para que haga de tope y prevenga que la tapa del contenedor, por fuerza centrífuga pueda desplazarse demasiado, desengranando el sistema y haciéndolo inservible aportando mayor seguridad al conjunto.

Los modelos se ajustarán a uno de los laterales del prototipo de modo que se aproveche la fuerza de la gravedad en la dispensación de los medicamentos, los cuales caerán fuera del contenedor. También se aprovecha el ángulo para prevenir que la fuerza centrífuga horizontal que no se vea mermada por la gravedad.

Como se trata de un dispositivo de pruebas, se emplearán tres modelos en el mismo, de modo que este primer construable tendrá capacidad para tres tomas básicas (desayuno, comida y cena),

## Módulo RTC (Reloj de tiempo real)



Como la división a lo largo del día de las tomas ha de ser, por propia definición, temporal, necesitaremos un método de cálculo del tiempo, para ello emplearemos un módulo RTC (real time clock - reloj de tiempo real), el cual simplemente es un reloj capaz de llevar un control tanto del día como de la hora, por lo que, como se aprecia, el grado de personalización que conseguimos ya es más elevado que la gran mayoría de soluciones que podemos encontrar en el mercado, pudiendo establecer pautas dependiendo del día y no solo de la hora.

El módulo elegido es el DS3231 y su funcionamiento es muy sencillo, la hora inicial se toma del ordenador cuando el Arduino está conectado al mismo, esta se mantiene cuando no está alimentado por el Arduino gracias a una batería propia. Una consulta nos permite obtener información de la fecha y hora y emplearlo ya sea para mostrarlo al usuario, almacenar algún tipo de registro o ejecutar secciones de código si se cumplen condiciones de carácter temporal.

## Motor

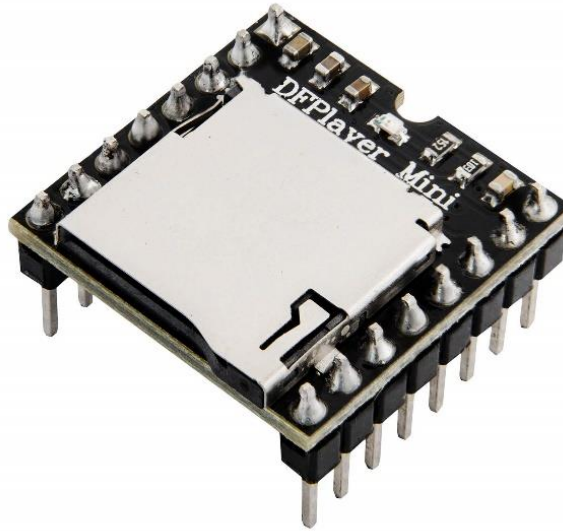


Como se comentó con anterioridad, se ha elegido un micro servo de 9g. La razón de la elección es puramente comercial, este es uno de los motores básicos incluidos en los kits de iniciación de Arduino que se pueden encontrar ya sea en tiendas o en páginas de compra online, económico, fácil de emplear, fácilmente adquirible y con gran cantidad de documentación disponible con una simple búsqueda por internet. Incluye además sus propios amarres que se pueden emplear en diferentes acoples a las construcciones que queramos llevar a cabo, los cuales nos serán sumamente útiles para asegurar los engranajes de los dispensadores que van insertados en la boca del motor.

Por el diseño de carga baja y el empleo de agentes lubricantes se podría emplear motores más pequeños para mover el dispensador, lo cual reduciría el volumen necesario para los modelos, no obstante, en una comparativa de precios estos encarecían el prototipo y se ha considerado apartar la idea de la incorporación relegándolos a una posible revisión o un futuro estipulado en las posibles mejoras del diseño actual



## Módulo de audio

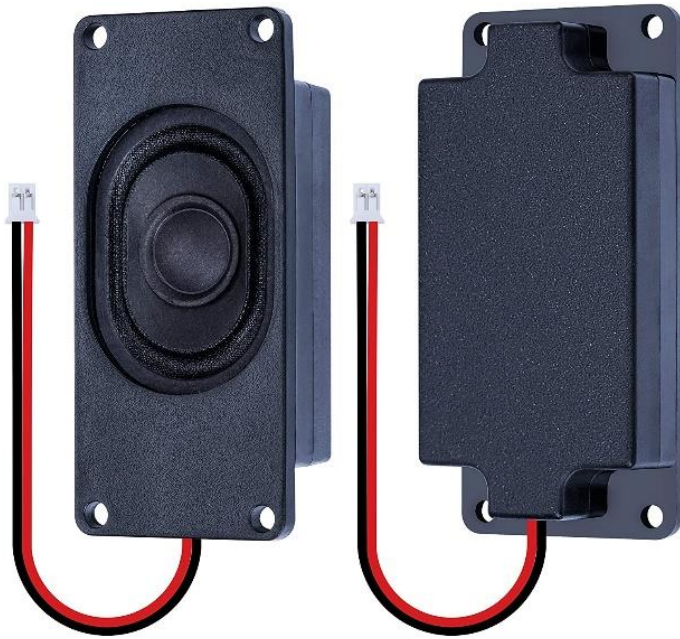


El audio es un punto muy importante y en el que se ha puesto especial atención. Para poder incluir audios personalizados, punto que hemos marcado como aspecto a incluir en el dispositivo, se ha elegido el Módulo DFPlayer mini, el cual permite reproducir sonidos contenidos en una tarjeta SD que se acopla al mismo, la comunicación se establece mediante el puerto serie UART del Arduino. Incluyendo en el sketch una librería propia para el módulo, ampliamente probada y documentada, podremos modificar el volumen para adaptarlo al paciente e incluso aplicar filtros al audio y seleccionar distintas fuentes, lo que nos permite un grado de personalización y adecuación elevado.

Para su inclusión en el prototipo, se ha soldado el módulo, así como los altavoces, de los cuales hablaremos un poco a continuación, a una pequeña placa de prototipado, de modo que sea más sencillo localizar la pieza e incluirla en el modelo final, que si bien buscamos que sea modificable y modular para poder efectuar cualquier cambio ya sea por adición sustracción o sustitución, no debemos incurrir en un montaje farragoso y poco amigable con el usuario.

Al módulo se le ha de conectar en el puerto rx una resistencia de 100 Ohmios. La conexión por el puerto serie afecta tanto a la calidad del audio como al funcionamiento del módulo, es por eso por lo que precisaremos de la conexión, a modo de “filtro” la resistencia, para evita estos efectos adversos.

## Altavoces



Los altavoces don de 8 Ohmios de resistencia y 3 Vatios de potencia, coincidiendo con los dispositivos de mayores características compatibles con el módulo de sonido, según las especificaciones del modelo.



La forma en la que se han colocado los altavoces es mediante un conector JST de dos pines, el cual se ha soldado a la antes referida placa. El tipo de conector ha sido decisivo a la hora de elegir estos altavoces, tras realizar a varias pruebas con el módulo, se ha comprobado que una mala conexión de los altavoces provoca que el módulo detenga la reproducción la música, por lo que la interacción con el pastillero, empleando una conexión poco segura, acortaría los mensajes, lo que supondría no cumplir uno de los objetivos que hemos marcado como importantes.

## Teclado matricial de membrana 4x4



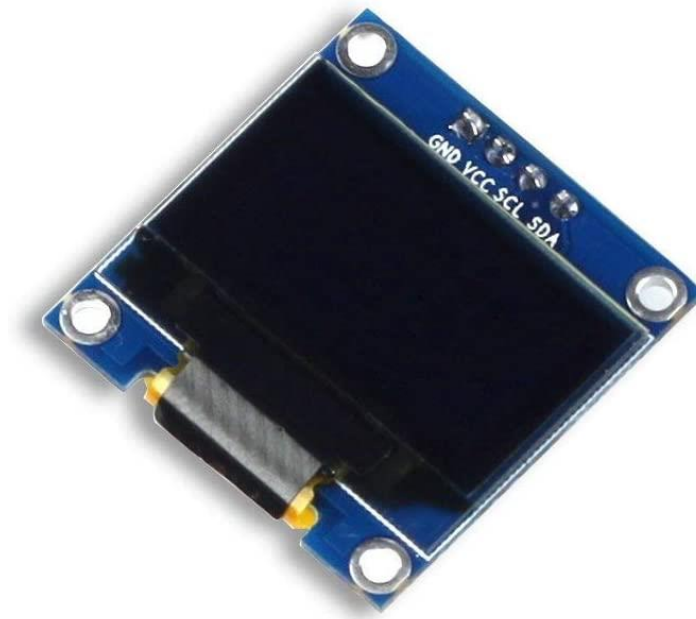
Emplearemos un sencillo teclado matricial, un módulo fácilmente implementable, que cuenta con amplia documentación y librerías que facilitan su uso y está muy testeadas como método sencillo de introducir un método de “diálogo” con el prototipo.

Si el objetivo del pulsador era el de ampliar la interacción del paciente, la introducción del teclado nos permite ampliar la del técnico a cargo del dispositivo. Con el pulsador podremos cambiar entre un modo de funcionamiento normal (como pastillero) y el modo de configuración que nos permitirá transmitir instrucciones sencillas al dispositivo.

Como muestra del rango de posibilidades que nos permite, veamos algunos de los códigos empleables y sus efectos en el dispositivo.

- #1A9 – Abre todos los contenedores.
- #1B9 – Cierra todos los contenedores.
- #1A<num. contendor> – Abre el contendor especificado
- #1B<num. contendor> – Cierra el contendor especificado
- #2A0 – Inicia la reproducción de la alarma.
- #2B0 – Finaliza la reproducción de la alarma.

## Pantalla OLED



Añadiremos, también, una pantalla oled, en este caso el SSD1306, el modelo es de 0.96 pulgadas (128x64 píxeles) pero, como sencillamente será empleada para mostrar la hora e información útil para el usuario, no necesitamos un modelo de mayores dimensiones.

La comunicación es I2C, lo que implica que deberemos emplear los pines SCL y SDA del Arduino para la conexión. De nuevo se trata de un módulo con gran cantidad de información disponible, así como de multitud de librerías que simplifican su uso.

La información que se muestra por pantalla ha sido en una gran parte de los casos, implementada por un proceso de prueba y error, queda para un futuro implementar una serie de funciones propias que controlen la forma en la que se muestra la información.

La pantalla contendrá la mayor parte del tiempo información sobre el día y la hora. Cuando llegue el momento de tomar los medicamentos, la pantalla mostrará una información adecuada a una señalización clara y evidente, mientras el sistema en su conjunto indica al paciente cual ha de ser el protocolo. También mostrará el código que se introduce en el modo de configuración y los resultados de los códigos insertados.

## ESP01-S



El módulo ESP01-S es una placa controladora con capacidades muy concretas y aplicaciones muy específicas. Con tan solo 6 pines, dos de ellos para la alimentación a 3.5 Voltios y dos para la comunicación serial por puerto UART, no tiene mucha capacidad de control simultáneo, pero se focaliza en el control de pequeños dispositivos, generalmente, apoyándose en su capacidad de conexión a internet.

Para hacernos una mejor idea de las capacidades de este módulo, ilustraremos con una de sus aplicaciones reales, la domótica. Incluyendo un relé al conjunto, podemos controlar, por ejemplo, las luces de nuestro hogar, enchufe o, incluso, un motor que pueda abrir y cerrar cortinas o persianas.

Por su reducido tamaño y su capacidad para conectarse a internet, por ejemplo, mandando una petición de tipo POST, nos servirá para darle mucha más utilidad al prototipo. El problema es su alimentación, ya que nos tocará disminuir el voltaje si queremos que nuestro puerto UART del Arduino pueda “dialogar” con el módulo.

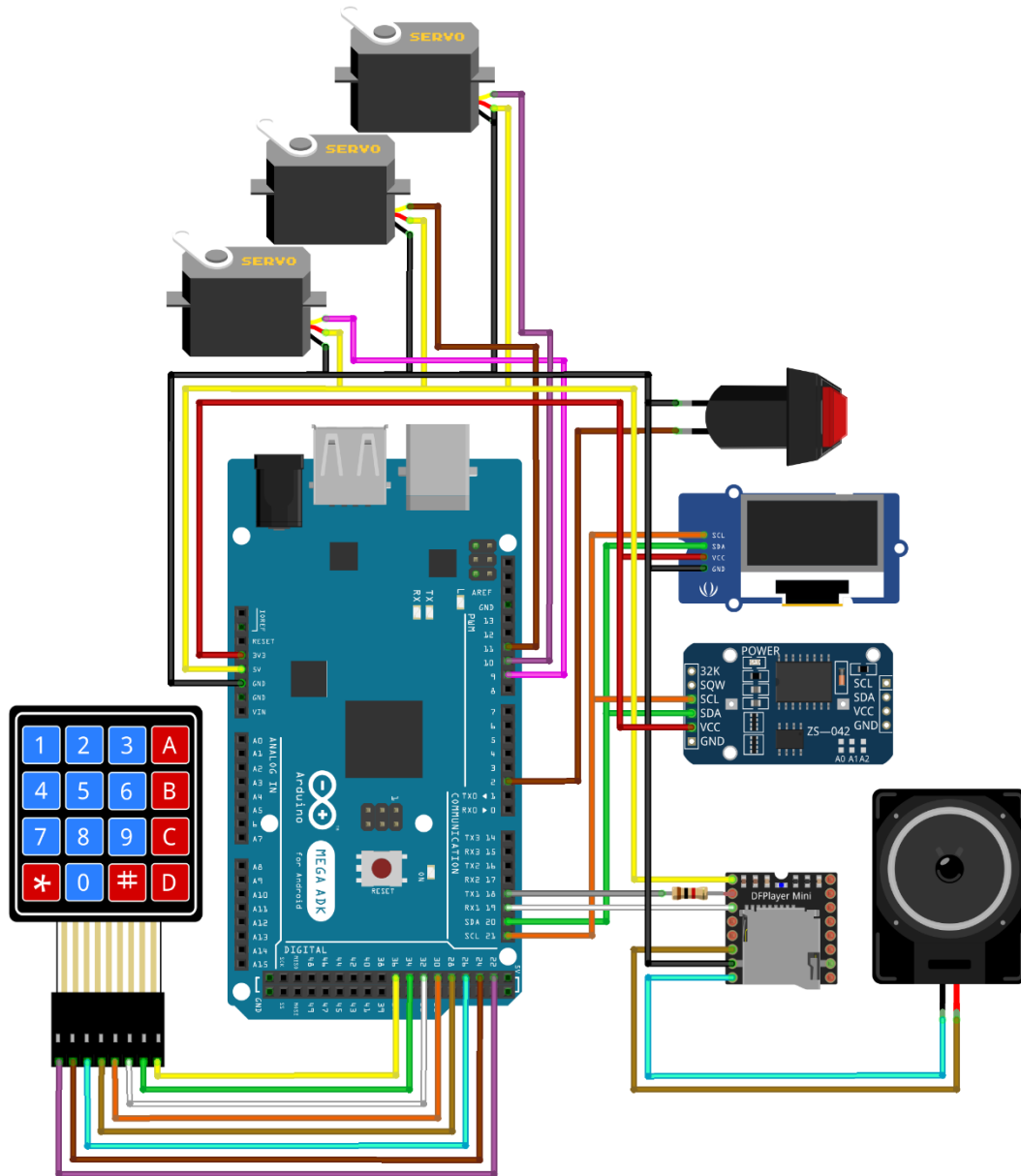
Si bien, este proyecto trata de aplicarse a una persona a la cual esta funcionalidad no le sirve, las ventajas que aporta poder conectarse a internet y enviar algún tipo de mensajeo feedback sobre las tomas de medicamentos es una funcionalidad nada desdeñable, es por eso, que, aunque de momento no se va a incluir en el prototipo si se va a desarrollar para una inclusión futura.

## Botón



Por último, un botón que sirva para elevar el nivel de interacción del paciente con el mecanismo. La idea es que funcione como accionador del proceso de dosificación siempre y cuando sea el momento adecuado, la alarma sonaría y para desactivarla se debe pulsar el botón para que se abra el mecanismo, una vez se han tomado las pastillas, se volvería a cerrar el dispensador. Esta manera de actuar nos permite, también tener retroalimentación de lo que ocurre a la hora de las tomas. A mayores, nos servirá para modificar el modo el dispositivo y, de esta manera, permitir a un posible cuidador interactuar con el dispositivo de forma cómoda.

## Modelo cableado



fritzing

Ilustración 15- Representación por bloques del esquema de conexiones

En la ilustración podemos ver el esquema de conexiones que se ha llevado a cabo para este proyecto, en este apartado entraremos a comentar tanto las decisiones que se han tomado sobre la elección de pines, los métodos de comunicación, las formas de alimentación y los modos de funcionamiento.

## Conexiones realizadas y pines en uso

Lo primero que vamos a comentar es el empleo del tablero de prototipado, mientras que un diseño acabado y completo nos exigiría el desarrollo de una placa de circuitos, el carácter experimental y de prototipo, nos hace decantarnos más por el empleo de esta tabla de prototipado con el objetivo de poder modificar las conexiones, añadir elementos y conectar o desconectar los diferentes módulos ya sea para llevar a cabo distintas pruebas o retocar el funcionamiento de alguno de ellos. El tablero de prototipado también nos permite introducir la fuente de alimentación de los carriles laterales, muy conveniente en estos primeros momentos donde nada es certero, puesto que puede no adaptarse a las necesidades, incluso aunque su funcionamiento sea no solo correcto si no el esperado, pero esto lo comentaremos más adelante.

Emplearemos los puertos 18 y 19 para hacer uso de la conexión por puerto serie declarada de forma independiente a la comunicación serie del Arduino con el ordenador por medio del conector USB, El módulo de sonido es el que empleará estos puertos.

Emplearemos el pin 2, puesto que programaremos una interrupción.

Emplearemos los pines del 9 al 11 para el control de los servomotores, estos pines tienen capacidad de uso de una señal PWM indispensable para el funcionamiento de estos.

Emplearemos los puertos 20 y 21 para la comunicación I2C necesaria para el display y el módulo RTC.

Para la conexión del keypad emplearemos los puertos pares del 22 al 34, ambos dos incluidos, el motivo de la elección de estos puertos radica en el tipo de conector del citado módulo, ocho cables diferentes uno a continuación del anterior, por lo que, como disponemos de una amplia variedad de pines por estar usando el Arduino Mega, elegiremos esta zona aislada y fácilmente conectable.

Emplearemos el pin de 3,5 Voltios,

Emplearemos los tres pines GND con los que cuenta este Arduino, uno para cablear directamente el botón, otro para el negativo de la alimentación de 3,5 Voltios del módulo RTC y la pantalla y otro para la conexión de las dos tierras, la del Arduino y la de la fuente de alimentación, con lo que conseguiremos la misma referencia entre ambas.



## Métodos de comunicación y control

### Comunicación I2C

El protocolo I2C (circuitos inter-integrados, en inglés *Inter-Integrated Circuit*), es un tipo de comunicación serial desarrollado para la comunicación de varios chips integrados de modo que fuesen parte de un todo, en el momento de su creación de un televisor de la compañía Phillips de la década de los 80. Lo que nació como un protocolo de comunicación dentro de un circuito “estanco”, sería desarrollado y adoptado por muchas otras compañías convirtiéndolo en un estándar de la industria y permitiendo su aplicación a otro tipo de circuitos integrados y placas que interactuaban con dispositivos que también contasen con la misma compañía aportando gran versatilidad.

El protocolo I2C funciona mediante una estructura de maestro-esclavo. Una arquitectura en la que uno de los dispositivos actúa como controlador y el resto como periféricos, siendo esta la distinción de ambos dos elementos que intervienen en la comunicación:

- Maestro o Controlador. Se trata del dispositivo que inicia y controla el proceso de comunicación. Este rol no es estático si no que es ocupado por el dispositivo que en cada momento “tome el testigo” e inicie la trasmisión de datos.
- Esclavos o Periféricos. Este rol es desempeñado por los dispositivos que no tienen que transmitir información al resto, estos quedan a la espera de que un maestro inicie la comunicación.

La principal ventaja de este protocolo frente a otros es la capacidad de conectar varios dispositivos al mismo tiempo, el controlador transmite un mensaje el cual es recibido por los demás, no obstante, no todos los mensajes tienen que afectar por completo a los periféricos, es posible manejar diferentes circuitos con diversos propósitos en cada momento.

Los mensajes son enviados mediante los pines SCA y SCL, el primero “escribe” los bits que incorporan la información, mientras que el segundo indica cuando ha de leerse el valor del primero.

Estos mensajes están compuestos por varios segmentos, indicando información tan relevante como el inicio y fin del mensaje, el periférico objetivo del mismo y varios datos como pueden ser órdenes, direcciones, etc.

## Protocolo UART

Se trata de un protocolo de comunicación en serie, es decir transmite la información como una secuencia de bits (1 y 0, de forma alternativa). Para llevar a cabo la comunicación se precisan el empleo de dos pines Rx para la recepción de datos y Tx para la transmisión, por lo tanto y como es esperable, para poder conectar dos dispositivos habrá que intercambiar los puertos, haciendo coincidir transmisores y receptores, limitando la conexión a un par.

Las placas de Arduino, casi en su totalidad, presentan, al menos, un puerto UART. Durante la comunicación se opera a nivel TTL de 0 a 5 Voltios en la transmisión de mensajes y reconocimiento de los bits, compatible con el uso de un USB, es por eso que la programación de las placas ya sea por medio del empleo del compilador del entorno de desarrollo de Arduino u otro tipo de métodos se realiza mediante esta vía y la aprovecha para establecer conexión con el puerto serie del mismo, con funcionalidades como el intercambio de mensajes, información, valores, etc.

BOARD	USB CDC NAME	SERIAL PINS	SERIAL1 PINS	SERIAL2 PINS	SERIAL3 PINS
Uno, Nano, Mini		0(RX), 1(TX)			
Mega		0(RX), 1(TX)	19(RX), 18(TX)	17(RX), 16(TX)	15(RX), 14(TX)
Leonardo, Micro, Yún	Serial		0(RX), 1(TX)		
Uno WiFi Rev.2		Connected to USB	0(RX), 1(TX)	Connected to NINA	
MKR boards	Serial		13(RX), 14(TX)		
Zero	SerialUSB (Native USB Port only)	Connected to Programming Port	0(RX), 1(TX)		
Due	SerialUSB (Native USB Port only)	0(RX), 1(TX)	19(RX), 18(TX)	17(RX), 16(TX)	15(RX), 14(TX)
101	Serial		0(RX), 1(TX)		

*Tabla 1 - Resumen de las conexiones series que presentan distintos modelos de Arduino*

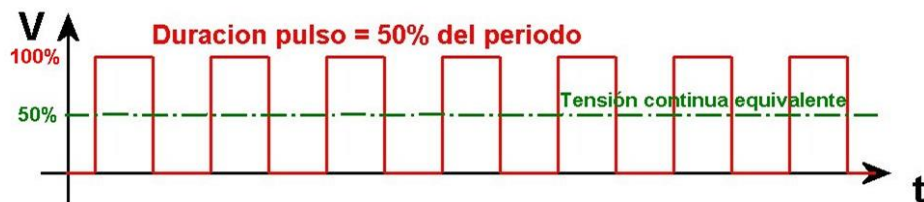
En la tabla podemos observar los pines puertos de que dispone cada Arduino y los pines que se emplean en su uso, en nuestro caso y como ya hemos apuntado, emplearemos el puerto serie 1 (Serial1), siendo el puerto serie por defecto el número 0 (Serial), el que se empleará para la conexión con el ordenador, cabe destacar la importancia de recordar a la hora de plantear aplicaciones con Arduino, que la conexión con el ordenador siempre se impone.

Nuestro Arduino Mega tiene cuatro puertos serie. En una placa que solo contase con uno, nos encontraríamos ante la problemática de solo poder emplear un puerto serie para, por ejemplo, la comunicación con un ordenador, lo que nos limitaría el uso y testeo del prototipo.

## PWM

Ya que las salidas digitales no pueden ofrecer valores específicos de tensión y, en muchas aplicaciones, se precisa de esta característica con un nivel de precisión adecuado. Para atajar esta limitación se puede emplear la modulación del ancho de pulso, una técnica que consiste en aplicar la máxima tensión disponible durante un tiempo dentro del periodo, de modo que, en cómputo global, durante ese periodo la media entre los valores “1” y “0” de como resultado el valor analógico que pretendemos obtener.

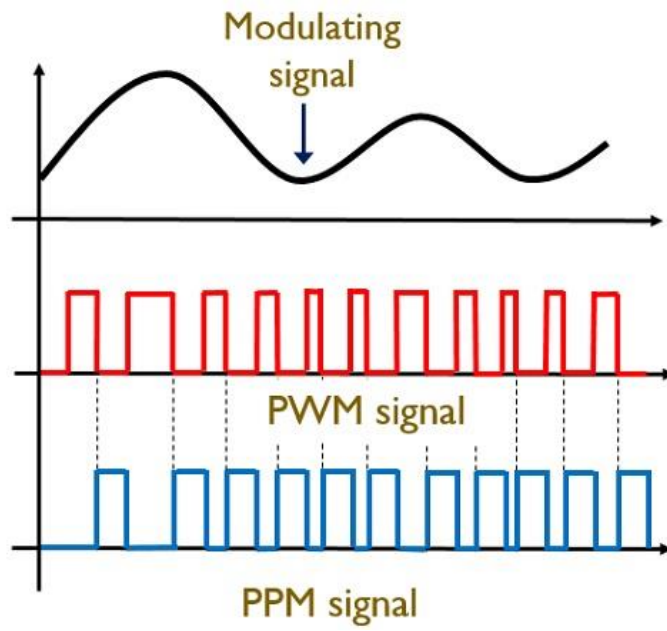
Durante la aplicación de esta técnica, es importante mantener la frecuencia, es decir, el tiempo entre activaciones, modificando la anchura del pulso, entendiendo esta como el tiempo en el cual en la salida tendremos un valor digital “1”, alto, o activo.



*Gráfica 2 - Tensión continua equivalente por alteración de la activación a lo largo de un ciclo*

El valor analógico resultante, como hemos comentado, se expresa como la media del estado de la salida en ese periodo concreto, por ejemplo, si la salida es activa a 5 Voltios y la mantenemos durante un quinto del tiempo, obtendremos un valor analógico para nuestra frecuencia de 1 Voltio, no obstante, la tensión que se suministra sigue siendo de 5 Voltios, luego esta técnica no sirve para alimentar un circuito que soporte una tensión menor que la del estado activo de la salida.

La modulación por ancho de pulso también nos puede servir, dependiendo de la aplicación, para modular una onda como se aprecia en la Gráfica 3 .



Gráfica 3 - Modulación de una señal mediante el uso de PWM

## Formas de alimentación

Hablemos ahora de las formas de alimentación que vamos a emplear. Con la idea en mente de que la pretensión es la de construir un prototipo en la que la estructura sea fácilmente modificable si se da el caso de que algún elemento no se desempeñe en la manera que debería.

El uso de una protoboard para el enganche de los motores se hace palpablemente necesario, si bien el resto de los elementos que empleamos para la implementación son comerciales, lo que implica que están ampliamente testeados y su funcionamiento bien descrito y acotado, los contenedores son de diseño propio y por ende pueden dar una variedad de problemas desde desgaste por el uso continuado, mal funcionamiento, necesidad de reemplazarlos en su totalidad o de forma unitaria, etc. Para la alimentación de la protoboard, se empleará de un adaptador que transforma la tensión de nuestros hogares a 5 Voltios. El empleo de una fuente de alimentación de 5V nos va a permitir desacoplar toda la parte de alimentación de los motores, pero no la del módulo DFPlayer, puesto que, por cómo funciona, si empleamos esta fuente de alimentación introduciremos ruidos a los motores y estos pueden empezar a moverse de forma descontrolada.

Por lo anteriormente comentado y ya que precisamos de una fuente de 3.3 Volios, emplearemos la alimentación de Arduino para alimentar los módulos que lo precisen. No podemos abusar mucho de este tipo de alimentación, pero, al tratarse de un Arduino Mega y como no estamos empleando muchas conexiones ni exigiendo demasiado rendimiento operacional al controlador, es asequible el poder usar estos dos pines.

Para alimentar la placa emplearemos una pila de petaca, o pila de 9 Voltios, también podremos alimentar el controlador mediante el ordenador mientras lo estemos usando, de este modo tendremos la capacidad de realizar operaciones de testeo y control. El motivo de emplear principalmente la pila es lo bien que se adapta a el espacio disponible y la filosofía del prototipo, pero en futuras revisiones la alimentación ha de tratarse de forma conjunta y con mayor detenimiento.

Por último, cabe comentar, que todas las tierras, es decir, el negativo de la fuente de alimentación y el pin GND del Arduino han sido conectadas, ya que, de otro modo, los motores no podrían funcionar puesto que la referencia de tensión nos está modificando el proceso y veremos que la señal “no llega” a los servomores, estos no se van a mover de la manera que se lo pedimos o, si tenemos ruido, se moverán deforma errática.

## Modos de funcionamiento

Por último, comentaremos brevemente los usos particulares de las conexiones que nos permite Arduino y vamos a emplear para mejorar el funcionamiento.

Como pequeño apunte, recordemos que los pines 9, 10 y 11, pese a ser salidas digitales, como precisamos que sirvan para regular la posición de los servos emplearán una señal modulada por ancho de pulso, de la que ya hemos hablado en el apartado anterior,

Por otro lado, el botón servirá para cambiar el modo de funcionamiento del dispositivo, por lo que deberemos entender su actuación dentro del conjunto como un elemento disruptivo de la conducta generalista del mismo, la forma en la que se traduce a las capacidades de la placa controladora, es empleando interrupciones. Una interrupción permite ejecutar una función dada una condición relativa a un pin al que enlazamos dicha interrupción. Para nuestro caso, lo que vamos a detectar, por ser lo más adecuado para un botón de tipo pulsador, es un cambio en el estado del pin, para ello vamos a definirlo como una entrada y vamos a asignarle una tensión inicial, es decir, realizaremos de manera intrínseca una conexión en pull-down, siendo necesario que uno de los extremos se conecte a tierra.

Habiendo comentado todo esto, podemos hablar los tres modos de funcionamiento del dispositivo:

- Modo normal: Durante este modo, simplemente vamos a actualizar la hora, a comprobar que ningún otro modo esté activo y controlar que si se cumplen o no las condiciones temporales de las tomas, que dispararían el control de estas.
- Modo de gestión: Estando activo este modo, se podrán introducir comandos mediante el keypad para controlar varias funciones del dispositivo.
- Modo de dispensación: Cuando llegue el momento de la toma, el presente modo servirá como estado intermedio que servirá al paciente para pasar del aviso a la dispensación e informar al dispositivo de que el medicamento ha sido ingerido.

## Arduino Mega



*Ilustración 16 - Arduino Mega*

### Características

El Arduino Mega monta un microcontrolador ATmega 2560 y un reloj de 16 MHz, operando con un voltaje de 5V.

La placa puede ser alimentada con un voltaje de entre 6 y 20 Voltios, pero no es recomendable, siendo los valores típicos entre 7 y 12 Voltios. Cuenta con 16 pines analógicos y 54 digitales que se pueden configurar como entradas o salidas, 15 de ellos con capacidad de modulación del ancho de pulso (PWM). En la Ilustración 16, podemos ver un diagrama de los pines que incorpora.

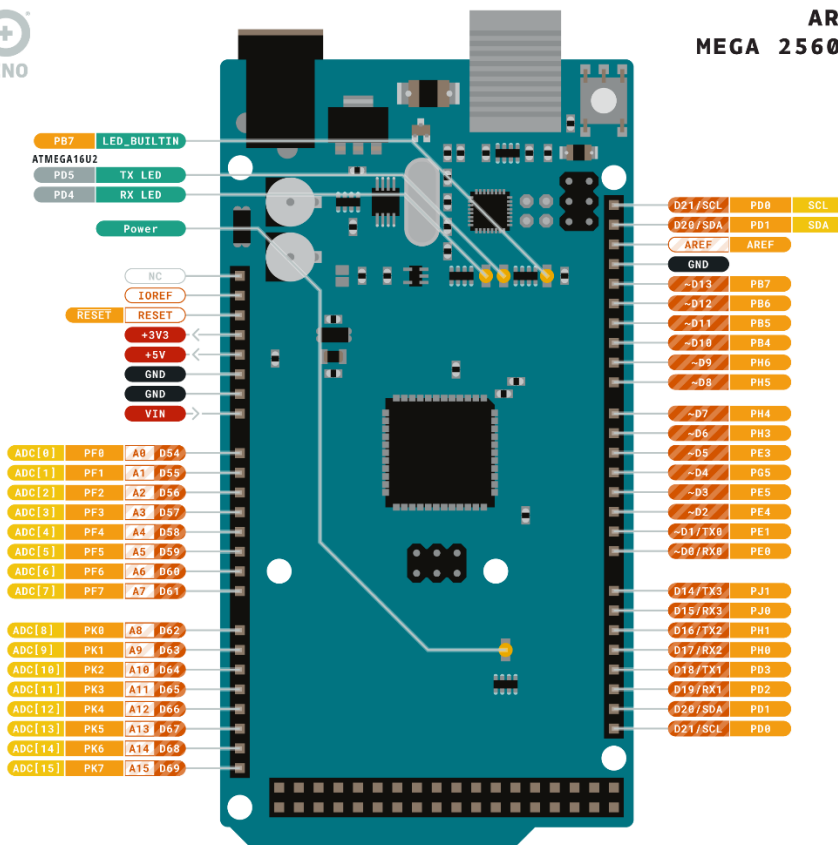
La corriente de salida es de 20 mA para todos los pines con excepción de la alimentación de 3.3 V, para la cual la corriente es de 50 mA.

Presenta tres tipos de memoria, un flash de 256 KB (8 KB de ellos empleados por el bootloader), una SRAM de 8 KB y una EEPROM de 4 KB. Es importante saber que cada escritura acorta la vida útil de la memoria y no es aconsejable emplearla de forma consistente.

Sus dimensiones son de 101.53 mm de largo, 53.3 mm de ancho con un peso de 37 g.



## ARDUINO MEGA 2560 REV3



Ground	Internal Pin	Digital Pin	Microcontroller's Port
Power	SWD Pin	Analog Pin	
LED	Other Pin	Default	

ARDUINO . CC  
  
This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/4.0/> or send a letter to Creative Commons, PO Box 1469, Mountain View, CA 94042, USA.

Figura 9 - Mapa de pines del Arduino Mega

### Justificación de la elección

Como se viene exponiendo a lo largo del documento, uno de los principales objetivos es la capacidad de adaptación del equipo a necesidades del paciente y potenciales mejoras, en este sentido, tener una gran cantidad de pines disponibles, además, con diferentes localizaciones y disposiciones nos aporta gran ventaja estratégica.

Por otro lado, el hecho de contar con varios puertos serie físicos facilita mucho el uso de este tipo de comunicación para la adhesión de diferentes módulos que precisen de esta. Esta característica puede ser suplida a nivel de código con el empleo de librerías como podría ser SoftwareSerial, pero requieren del uso sincronizado restando velocidad de uso.



## Software creado

Una de las necesidades que el prototipo debe satisfacer es la de contar con un sistema de configuración que sea simple pero completo. Para ello se ha desarrollado una interfaz gráfica que facilite la interacción del usuario con todos los elementos programables. En este sentido, la tradicional programación del controlador se ha visto acompañada tanto de una base de datos que permitiese almacenar toda la información necesaria, lo que expande el número de prototipos, pertenecientes a pacientes diferentes, que un mismo terminal puede llegar a manejar de forma eficiente, como un programa de edición que permita tanto interpretar los datos almacenados en la base de datos, como comunicarse con el controlador.

## Base de datos del prototipo

Antes de entrar a describir la interfaz gráfica, analizaremos brevemente la base de datos que sirve de esqueleto para la misma, pues esto permitirá entender mejor las interrelaciones subyacentes entre las distintas partes editables de la interfaz. El esquema de la base de datos sería el siguiente:

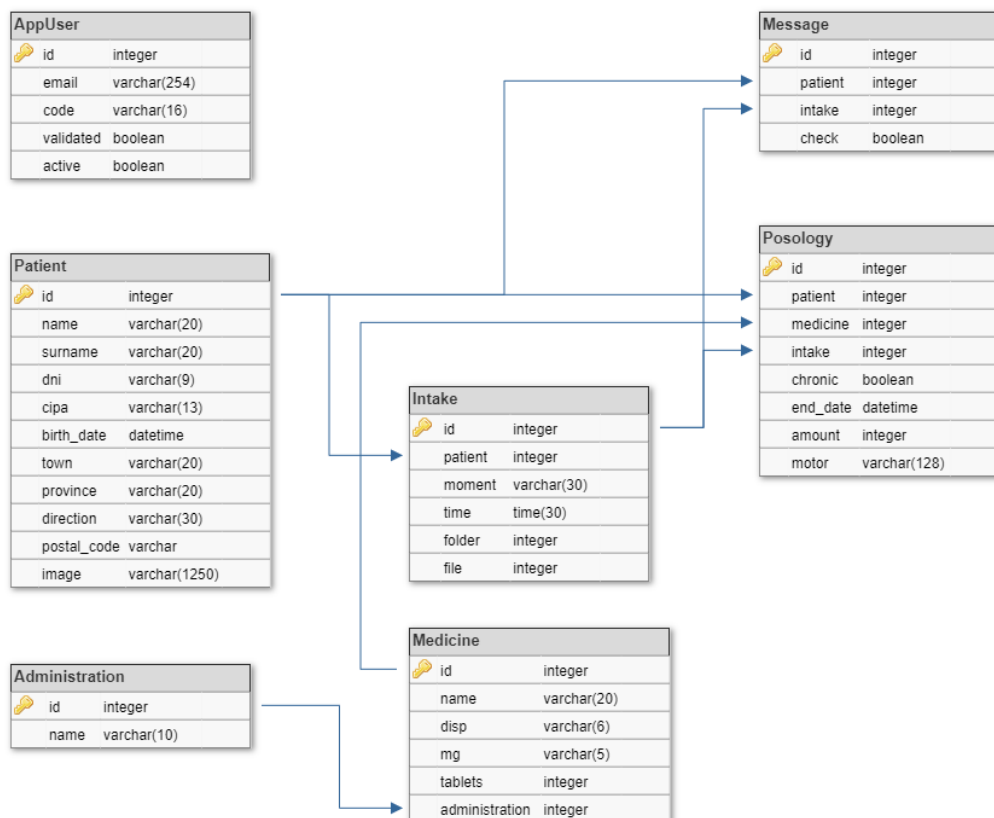


Figura 10 - Esquema de la estructura de la base de datos

En nuestro sistema sanitario, los pacientes disponemos de unas hojas de medicación, las cuales recogen los datos de los que ya disponían los médicos de cabecera a la hora de prescribirnos un tratamiento y que vienen a ser el sucesor de las archiconocidas “recetas médicas”, desde que los medicamentos pueden ser adquiridos en farmacia mediante la tarjeta sanitaria. Los documentos de medicación no solo recogen datos sobre los pacientes a los que van dirigidas de distinta índole, además, detallan la posología, o dosificación de estos, a fin de que el paciente tenga una visión holística o ayudar a las personas encargadas de sus cuidados, en caso de que precisen de ayuda externa.

Nuestra base de datos se ha creado con el objetivo de aunar la posología de las anteriormente citadas hojas de medicación, así como al resto de datos, permitiendo el rápido acceso de los cuidadores a todos ellos en función del paciente seleccionado. También se han incluido otros campos para dotar de mayor funcionalidad al prototipo.

Si hablamos de bases de datos relaciones, el lenguaje de programación SQLite, por su rapidez, sencillez, capacidad de trabajo con otros lenguajes de programación y porque el hecho de que no permita conexiones de varios usuarios no es un problema para el propósito de este prototipo, parece ser la elección más adecuada. El editor elegido es *DB Browser for SQLite*, software de uso libre ampliamente testeado y que sigue mejorándose periódicamente, a mayores, incluye herramientas para aquellos usuarios legos en el lenguaje de programación que no influyen en el potencial de personalización y agilizan los procesos de creación de tablas.

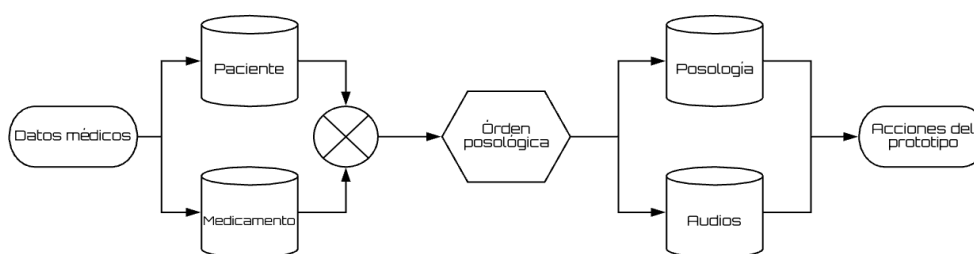


Figura 11 - Esquema de tratamiento de datos

En la anterior figura, podemos ver el proceso que llevaríamos a cabo a la hora de registrar los datos necesarios para almacenar tanto los datos de los pacientes como de los medicamentos, combinándolos para constituir órdenes posológicas transmitibles al prototipo, generando acciones que las desencadenen.

Para mejorar el rendimiento de esta base de datos, aplicando las posibilidades que nos aporta la librería elegida para crear la aplicación web, de la que hablaremos más adelante, importaremos la base de datos creada y reformularemos algunos de los campos limitados por el tipo de aplicación empleada. Con el objetivo de comprender un poco mejor esta base de datos, vamos a repasar las tablas que la componen explicando para cada caso los valores y deteniéndonos de manera específica en aquellos que han aplicado alguna de las funcionalidades que relatamos.

Para controlar las cuentas de los usuarios que empleen la aplicación web, se ha creado la tabla:

AppUser		
<b>id</b>	AutoField	Clave primaria, se trata del identificador de los registros para esta tabla
<b>email</b>	EmailField	Campo propio de la clase con constructores especiales en el formato para un email
<b>code</b>	CharField(255)	Contraseña creada por el usuario, a la que se le ha aplicado seguridad (desarrollada abajo)
<b>validated</b>	BooleanField	Campo booleano para indicar si la cuenta ha sido validada por el usuario.
<b>active</b>	BooleanField	Campo booleano para indicar si la cuenta ha sido aprobada por el super-administrador

Tabla 2 - Tabla AppUser de la base de datos

Para la contraseña de usuario, se ha creado un *validator*, es decir, una función que comprueba los parámetros típicos de una contraseña con un mínimo de seguridad (Inclusión de números y letras, de mayúsculas y con una longitud no inferior a 8 caracteres).

A mayores, para aumentar la seguridad, se ha establecido una encriptación de la contraseña, de modo que esta se almacena una vez convertida. El proceso

de acceso emplea el mismo método, se introduce la contraseña, se transforma y analiza en el lado del servidor y se devuelve el error o acierto.

Para controlar los tipos de administración se ha creado una tabla heredada de la construcción de SQLite, y que modificaremos en otras aplicaciones para registros homólogos, pero que hemos mantenido por dar heterogeneidad al código y por trabajar con los datos de varias maneras.

Administration		
<b>id</b>	AutoField	Clave primaria, se trata del identificador de los registros para esta tabla
<b>name</b>	CharField	Campo que contiene nombres fácilmente identificables y que diferencias los tipos de administración (ocular, oral, ...)

*Tabla 3 - Tabla Administration de la base de datos*

Para el control de las tomas de los usuarios en aquellos modelos del dispositivo que tengan capacidad de retroalimentación mediante el módulo ESP-01S, que enviaría una petición POST contamos con la tabla:

Message		
<b>id</b>	AutoField	Clave primaria, se trata del identificador de los registros para esta tabla.
<b>patient</b>	ForeignKey	Clave foránea que nos redirige a la tabla de Patient. Identifica al paciente que ha de realizar la toma del medicamento.
<b>intake</b>	ForeignKey	Clave foránea que nos redirige a la tabla de Intake. Identifica la ingesta a la que se refiere el registro.
<b>check</b>	BooleanField	Campo booleano para indicar si se ha tomado o no el medicamento

*Tabla 4 - Tabla Message de la base de datos*

Para tener un registro de los posibles medicamentos que vamos a poder asignar a los pacientes se ha creado la tabla:

Medicine		
<b>id</b>	AutoField	Clave primaria, se trata del identificador de los registros para esta tabla.
<b>name</b>	CharField(20)	Campo con el nombre del medicamento. Como un mismo tipo de medicamento puede tener una u otra concentración, se ha extraído esa información y no se debe incluir en el nombre.
<b>disp</b>	CharField(6)	Campo que contiene el código de identificación farmacéutico. Este puede ser empleado en la farmacia para identificar el mismo o por el gestor del dispositivo para discernir las cajas sin atisbo de duda.
<b>mg</b>	FloatField(5)	Campo en coma flotante para almacenar los miligramos de medicamento que contienen las pastillas. Pudiendo diferenciar diferentes concentraciones,
<b>tablets</b>	IntegerField	Campo de entero que indica el número de tabletas que contiene cada paquete del medicamento en cuestión.
<b>administration</b>	ForeignKey	Clave foránea que nos redirige a la tabla de Administration. Indica el tipo de administración del medicamento.

*Tabla 5 - Tabla Medicine de la base de datos*

Para la gestión de los pacientes, se ha creado la tabla:

Patient		
<b>id</b>	AutoField	Clave primaria, se trata del identificador de los registros para esta tabla.
<b>name</b>	CharField(20)	Campo con el nombre del paciente.
<b>surname</b>	CharField(30)	Campo con los apellidos del paciente.
<b>dni</b>	CharField(9)	Campo con el documento nacional de identidad del paciente.
<b>cipa</b>	CharField(13)	Campo con el número sanitario de identificación el paciente
<b>birth_date</b>	DateField	Campo con la fecha de nacimiento del paciente.
<b>phone_number</b>	IntegerField	Teléfono de contacto del paciente.
<b>town</b>	CharField(20)	Población donde reside el paciente.
<b>province</b>	CharField(20)	Provincia donde reside el paciente.
<b>direction</b>	CharField(30)	Dirección de la residencia del paciente.
<b>postal_code</b>	IntegerField	Campo con código postal del paciente
<b>image</b>	CharField(1250)	Campo que contiene la ubicación del archivo de imagen de perfil del paciente.

Tabla 6 - Tabla Patient de la base de datos

Cada paciente puede tener unos hábitos que marcan el horario que sigue, esto también afecta a los momentos de las tomas y para poder tener un control preciso de este aspecto contamos con la tabla:

Intake		
<b>id</b>	AutoField	Clave primaria, se trata del identificador de los registros para esta tabla.
<b>patient</b>	ForeignKey	Clave foránea que nos redirige a la tabla de Patient. Indica el paciente al que se refieren los horarios que se declaran.
<b>moment</b>	CharField(30)	Nombre de la ingesta. Este campo es meramente estético, ya que podría ser suplantado por una especificación temporal, pero, para facilitar la programación a los gestores del dispositivo, se ha decidió dotar a los registros de un nombre identificativo.
<b>time</b>	TimeField	Campo temporal que contiene el momento de la toma.
<b>folder</b>	IntegerField	Campo que contiene la identificación de la carpeta si queremos asociar un audio personalizado
<b>file</b>	IntegerField	Campo que contiene la identificación del archivo si queremos asociar un audio personalizado

*Tabla 7 - Tabla Intake de la base de datos*

Cada vez que se registra un paciente se crean 7 ingestas con tomas comunes y horas definidas siguiendo el horario típico en España. Estas entradas son modificables, también se puede añadir nuevas ingestas personalizadas.

Los campos *folder* y *file* se han creado para poder incluir audios personalizados en cada ingesta, ambos ubican el archivo de audio que se ha de reproducir.

Para llevar a cabo el control de las tomas se ha creado la tabla:

Posology		
<b>id</b>	AutoField	Clave primaria, se trata del identificador de los registros para esta tabla.
<b>patient</b>	ForeignKey	Clave foránea que nos redirige a la tabla de Patient. Indica el paciente al que se refieren la posología.
<b>medicine</b>	ForeignKey	Clave foránea que nos redirige a la tabla de Medicine. Indica el medicamento que ha de tomar el paciente.
<b>intake</b>	ForeignKey	Clave foránea que nos redirige a la tabla de Intake. Indica el momento de la ingesta del medicamento.
<b>chronic</b>	BooleanField	Campo booleano para indicar si la toma es crónica o no, es decir, si siempre ha de hacerse.
<b>end_date</b>	DateField	Campo obligatorio para medicamentos que no son crónicos, indica la fecha a partir de la cual ya no es necesaria la toma del medicamento.
<b>amount</b>	IntegerField	Campo que indica la cantidad de pastillas que se ha de ingerir.
<b>motor</b>	CharField	Campo que indica el servomotor que se ha de activar para depositar la pastilla.

*Tabla 8 - Tabla Posology de la base de datos*

Cabe destacar que el campo motor obliga a elegir una de las opciones disponibles, es otra forma de implementar un registro que solo tiene como campos un identificador y un nombre, como es el caso de la tabla *Administration*, este método, para pocas entradas es muy útil.



## Aplicación WEB

### Lenguaje de programación, *framework* y módulos

El lenguaje de programación elegido para llevar a cabo el desarrollo de la aplicación web es Python. Se trata de un lenguaje de programación ágil, que nos permite tanto un proceso de aprendizaje rápido como una reducción del tiempo necesario para la creación de funciones que conformen el esqueleto de la aplicación en relación al que sería necesario empleando lenguajes de programación de mayor nivel de abstracción. Python es un lenguaje de programación con una implementación en auge y muchas empresas deciden construir sus aplicaciones basándose en este lenguaje, tenemos casos notables como Instagram, Google, Netflix o Spotify entre otros.

Una aplicación web necesita un marco en el que construirse, podríamos emplear nuestro propio *framework*, pero parece más inteligente emplear uno ya existente, testado y robusto. La elección en este sentido ha sido Django. De nuevo, estamos ante un marco de trabajo para Python de alto nivel, con la capacidad de desarrollo de sitios y aplicaciones web de forma rápida y cómoda. También tenemos casos notables de uso como las páginas web de The New York Times, Pinterest, NASA Science, The Washington Post, National Geographic o Mercedes-Benz entre otras.

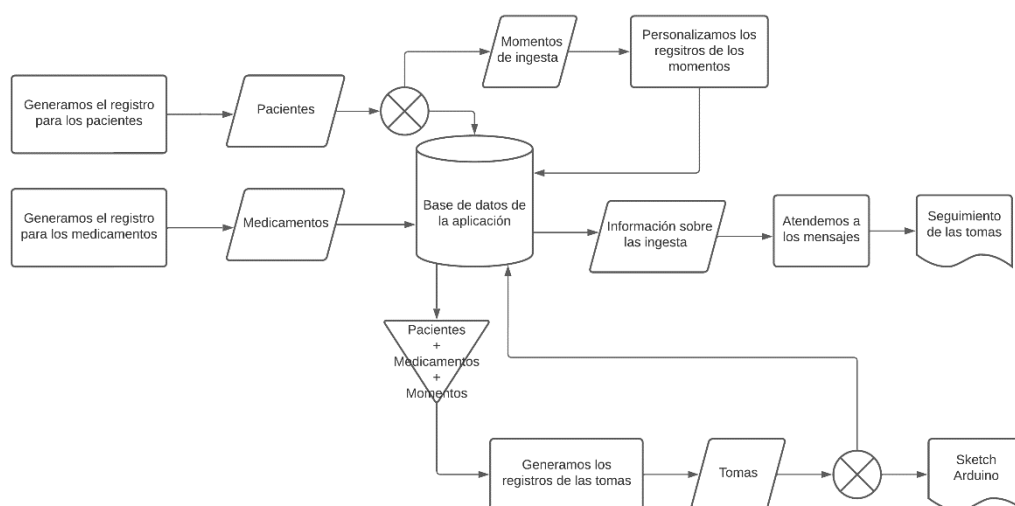
El motor de la aplicación no lo es todo, pues una aplicación poco vistosa y amigable con el usuario, por potente que sea, pierde parte de su capacidad y calidad, por eso para el *front-end*, nos ayudaremos del marco de trabajo que nos aporta Materialize, basado en Material Design, es un conjunto de librerías tanto de CSS como de JavaScript que nos ofrecen herramientas básicas sobre las que estructurar una página web y que pueden ser editadas para actualizar el estilo a nuestras conveniencias, como es lógico emplearemos lenguaje HTML para el diseño de páginas, complementándolo visualmente con el diseño que nos permite CSS y terminando el pack con las funcionalidades que JavaScript nos permite conseguir.

Para poder implementar la conexión con la aplicación mediante terceros, emplearemos el marco de trabajo de REST Framework, que nos permite programar una serie de accesos siguiendo y protocolo REST (con peticiones del tipo GET, POST, PUT y DELETE) para conectarse mediante lo que se conoce como una API.

Por último, incluiremos el módulo cryptography, para poder incluir algo de seguridad con la inclusión de un encriptado de las contraseñas de acceso.

## Esquema de uso

En la siguiente figura, se intenta resumir las posibilidades de la aplicación web que comentaremos en el presente apartado, se ha intentado construir de izquierda a derecha en un orden lógico atendiendo a los sucesos que se han de dar de forma temporal, si bien una aplicación web no tiene por qué ser empleada siguiendo los pasos de su concepción, nos ayudará en la explicación.



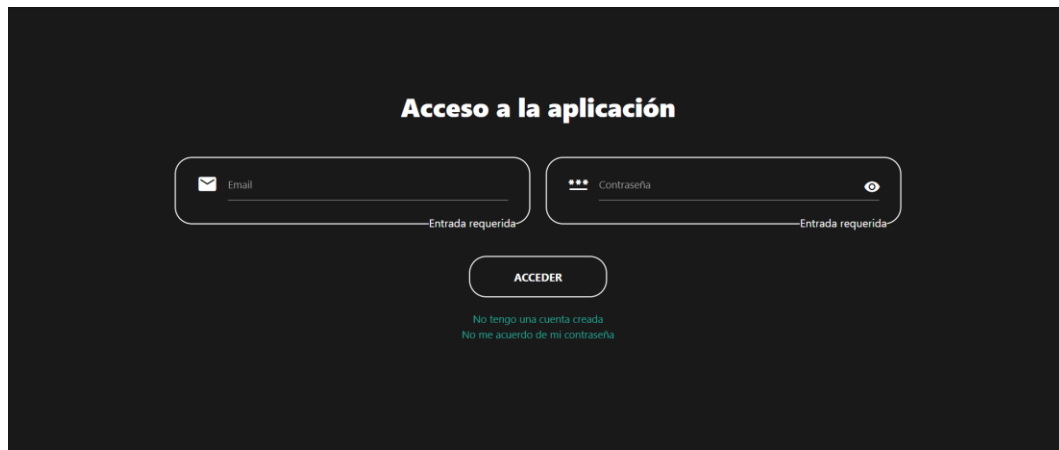
*Ilustración 17 - Esquema de uso de la aplicación*

Cabe destacar que, la pieza angular que vertebra el trabajo compone el centro del esquema y se ha explicado en primera instancia debido a su peso en el desarrollo es la base de datos. La base de datos es el punto de unión de todos los procesos de nuestra aplicación. Todas las acciones, crean, modifican, muestran o eliminan los registros de esta y es a partir de ellos que se obtiene tanto los dos documentos de control como el esquema del pastillero del paciente.

Lo primero que debería hacer el gestor del dispositivo o de la aplicación web es generar los registros correspondientes a los pacientes y los medicamentos. Esto es clave ya que cada momento de la ingesta tiene asignado un paciente y cada toma declarada tiene asignado uno de los posibles valores de los ya comentados tres registros. Los momentos de toma son totalmente personalizables, adaptándolos a los hábitos del paciente. Por último, se deben crear las tomas del paciente, para poder generar el sketch del Arduino, un documento que se puede pegar directamente en el IDE de Arduino para programar el mismo. Las ingestas, si el dispositivo cuenta con feedback, pueden controlarse mediante los mensajes.

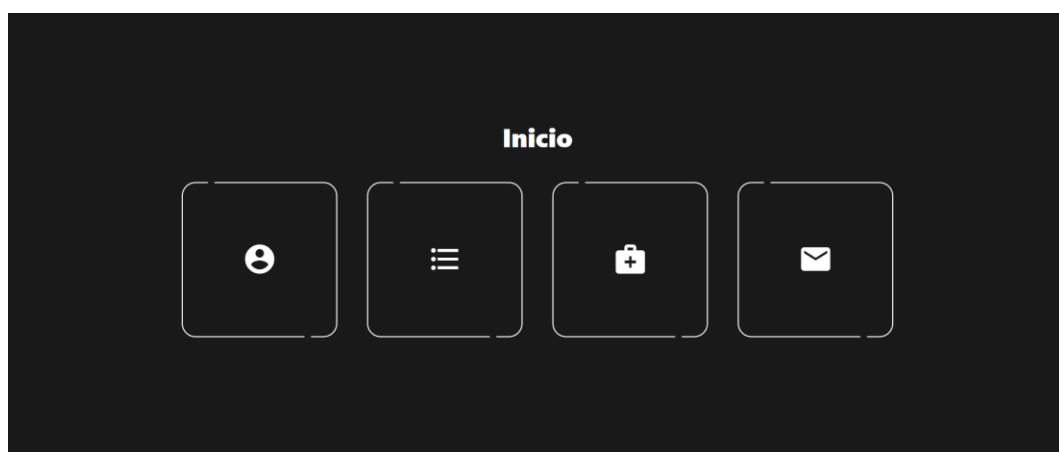
## Apartados de la página web

En este apartado pasaremos a comentar de forma más detallada las distintas partes que componen la aplicación web, indicando en cada caso las posibilidades que nos ofrecen y el proceder para explotar las mismas.



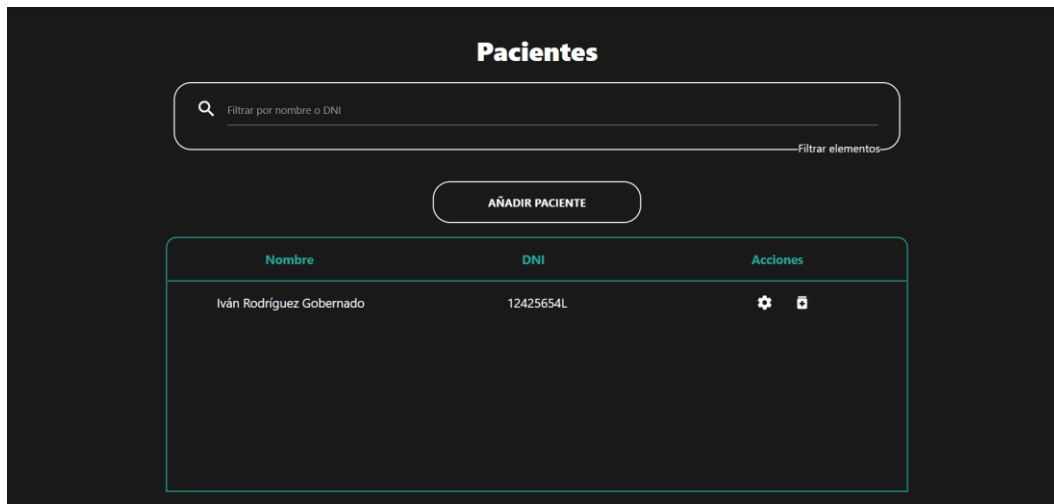
*WebApp 1 - Pestaña de acceso*

En la captura WebApp 1, podemos ver la página de acceso, la cual nos pide las credenciales para el acceso a la aplicación, cabe destacar que todas las cuentas han de ir vinculadas a un correo para mayor seguridad y que en ningún caso se guardan las contraseñas, se emplea una tabla hash que almacena las mismas encriptadas. Se ha implementado un diseño minimalista y se han incluido algunas funcionalidades como el cambio de contraseña entre el modo seguro que emplea puntos en vez de los caracteres de escritura y el modo texto que nos muestra lo que hemos escrito, así como dos opciones, una para crear una cuenta y la otra para recuperar la contraseña si es que no nos acordamos de la misma. La opción de cambiar la contraseña no existe por añadir capas de inseguridad.



*WebApp 2 - Página principal*

La captura WebApp 2, nos muestra la pantalla principal o menú de navegación de la aplicación, es la primera en mostrarse para ayudar a ubicarse al gestor desde un inicio. Nos muestra los cuatro apartados operativos principales de nuestra aplicación: Pacientes, Medicamentos, Pastilleros y Mensajes. En lo restante del presente apartado veremos las funcionalidades de cada uno de estos.



*WebApp 3 - Apartado de pacientes*

La captura WebApp 3, nos permite ver el apartado de pacientes, el cual nos permite crear los registros para los mismos, editarlos o borrarlos, en la tabla se van a mostrar todos los pacientes registrados y las dos funciones disponibles nos permiten la edición y el acceso al apartado del pastillero del paciente, pero no nos adelantemos.

Este apartado incluye una barra de búsqueda que nos permite filtrar por DNI o nombre, descartando aquellos pacientes cuyos datos no se corresponden con los requeridos.

Podemos ver en la captura **¡Error! No se encuentra el origen de la referencia.** el formulario que habrá que cumplimentar para registrar un nuevo paciente. Los datos son los que comentamos en la base de datos y se emplea por defecto una imagen para todos los usuarios que puede ser modificada en cualquier momento con una fotografía del paciente. Básicamente tenemos información personal y de contacto del paciente, así como documentos que lo identifican

## Crear nuevo registro de pacientes



No se seleccionó ningún archivo

T

Nombre

Tt

Apellidos



DNI



CIPA

Fecha de nacimiento



dd/mm/aaaa



Provincia



Localidad



Código postal



Dirección

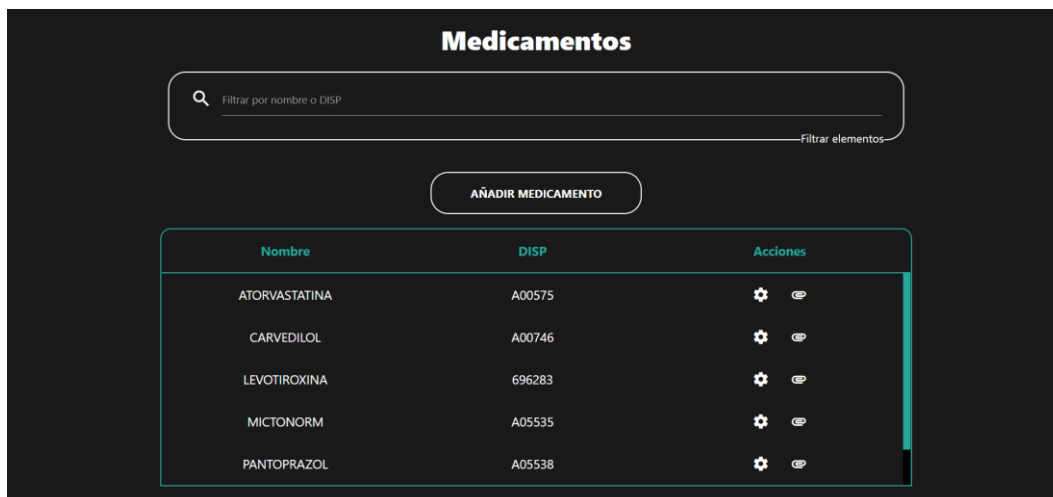


Teléfono

GUARDAR

CERRAR

WebApp 4 - Formulario para el registro de un nuevo paciente



*WebApp 5 - Apartado Medicamentos*

El apartado de medicamentos, que podemos apreciar en la captura WebApp 5, permite la gestión de los mismos de forma análoga a como lo hacíamos con los pacientes, podemos crearlos, editarlos o borrarlos. Los medicamentos registrados aparecerán en la tabla y las opciones que esta nos ofrece son la de edición y copia del código de modo que se pueda implementar de forma más sencilla en la pestaña de tomas, simplemente deberemos hacer clic en el botón que hemos comentado y dirigirnos a la página de toma de medicamentos para que, con un simple clic, podamos seleccionar de manera automática el medicamento del que copiamos el código.

De nuevo, esta vista cuenta con un buscador, esta vez el campo de filtrado es el nombre del medicamento y el “DNI” del mismo, es decir, del disp.

En la captura podemos apreciar el formulario que habremos de rellenar para poder incluir un nuevo registro de medicamento. La información que habría que cumplimentar es la referida en la base de datos.

Cabe destacar que todos los medicamentos que en esta pestaña se registren estarán disponibles para todos los pacientes de la plataforma, esto es así ya que muchos de ellos van a ser compartidos puesto que su objetivo o bien es el de combatir enfermedades comunes o bien palian los efectos del envejecimiento y las disfunciones corporales consecuentes, por ello, no sería lógico personalizar hasta tal grado que debamos introducir una y otra vez el mismo medicamento.

## Crear nuevo registro de medicamentos

T

<>

⚖

⊕

🧴

GUARDAR

CERRAR

*WebApp 6 - Formulario para el registro de un medicamento*

## Seleccionar Paciente

🔍
Filtrar elementos

Nombre	DNI	Acciones
Iván Rodríguez Gobernado	12425654L	🔗

*WebApp 7 - Apartado de Pastilleros (Selector)*

El apartado de pastilleros ilustrado con la captura WebApp 7, se sostiene en una página equivalente a una versión reducida del apartado de pacientes, esto es así ya que primero deberemos localizar al paciente y luego seleccionar su pastillero. Como se aprecian las funciones de búsqueda y acceso al pastillero son las únicas que se heredan del comentado apartado.



*WebApp 8 - Apartado de Pastilleros (personal)*

Este apartado que vemos en la captura WebApp 8, ya sí que corresponde con el pastillero personal del paciente, en el mismo, podremos crear, editar y borrar las diferentes tomas de medicamentos que queramos asignar al paciente.

Las diferentes tomas registradas, aparecerán en la tabla y pueden filtrarse tanto por la ingesta a la que correspondan como por el nombre del medicamento. Las opciones de cada son la de edición y gestión de su actividad de manera rápida, es decir, pueden desactivarse del pastillero, pero para poder activarse deberán editarse desde el formulario.

En este apartado tenemos dos nuevas opciones al lado del botón de creación de registros. La primera opción nos permite abrir el editor de momentos de ingesta, del que hablaremos posteriormente y la segunda opción nos va a generar un documento descargable que contiene el sketch Arduino, este puede ser empleado para cagar el programa del funcionamiento en el dispositivo si nos lo llevamos al IDE de Arduino y conectamos la placa.

El formulario que ha de cumplimentarse para generar estos registros aparece en la captura WebApp 9, de nuevo, los datos son los referentes al esquema definió en la base de datos.



### Crear nueva toma de medicamentos

Medicamento

+

Medicamento

🕒

Comprimidos

🕒

Fecha final de consumo

📅

📅

Medicamento crónico

Contenedor

📁

GUARDAR

CERRAR

*WebApp 9 - Formulario para el registro de tomas*

### Momentos para la ingesta

AÑADIR MOMENTO

Etiqueta	Hora	Acciones
Amanecer	08:00	⚙️
Desayuno	08:30	⚙️
Almuerzo	12:00	⚙️
Comida	14:00	⚙️
Merienda	18:00	⚙️

*WebApp 10 - Gestor de Momentos de la ingesta*

La captura WebApp 10 muestra el gestor de momentos de ingesta, mientras que la captura WebApp 11 nos ilustra el formulario que se ha de cumplimentar para crear los registros. Este apartado permite la creación edición y borrado de este tipo de entradas para las tablas de la base de datos. Ya hablamos de forma extensa de que son y para qué se crean estos datos así que no nos extenderemos más.

## Crear nuevo momento de ingesta

🏷️ Etiqueta

🕒 Hora

📁 Carpeta

📁 Archivo

GUARDAR

CERRAR

*WebApp 11 - Formulario para el registro de momentos de ingesta*

Para poder visualizar los mensajes que llegarían desde el pastillero acerca de si el paciente ha tomado o no los medicamentos, se ha implementado un apartado que puede apreciarse en la captura WebApp 12.

## Mensajes

🔍 Filtrar por Paciente o Ingesta 
—filtrar elementos—

Nombre	Ingesta	Estado
Iván Rodríguez Gobernado	Amanecer	✔️
Iván Rodríguez Gobernado	Amanecer	❌

*WebApp 12 – Apartado de mensajes*

## Pastillero Electrónico de Programación Individual

ACCEDER

### ¿Qué es?

Se trata de un prototipo de pastillero electrónico automático desarrollado como Trabajo de Fin de Grado que busca facilitar la vida de las personas que padecen Alzheimer.

Su diseño modular permite una rápida adaptación a las necesidades, no se trata de implementar un sistema cerrado, cada paciente tiene sus propias necesidades y la solución ha de ser capaz de satisfacerlas.

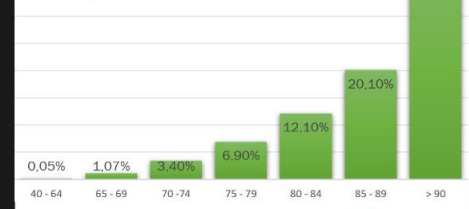
Buscando una implementación sencilla, amigable con el usuario y que evite ser un elemento disruptor en la rutina y la vida del paciente, se ha llegado a una solución multidisciplinar en la que se ha trabajado desde varios ámbitos para obtener una respuesta integral al problema.

En la imagen podemos ver el prototipo físico y esta aplicación web actúa como administrador del mismo.



### ¿Por qué?

#### Prevalencia de la Demencia por tramos de edad



La demencia es un problema que se puede llegar a manifestar con la edad. El aumento de la esperanza de vida junto con el empeoramiento de los hábitos mentales saludables que se ha instaurado y se consolida cada vez más en nuestra sociedad, el Alzheimer se ha convertido en un problema de orden internacional.

Las demencias en su conjunto plantean un problema sociosanitario, pues no solo empeoran la calidad de vida de quien las padecen, también suponen un reto para los familiares que se encuentran ante una situación para la que no tienen preparación ni en términos de cuidados, ni psicológicamente. El grado de dependencia es, como se puede esperar, privativo y opresivo. La principal vía de actuación y que, con este proyecto, se plantea es la de recuperar parte de la autonomía de la persona que podrá disfrutar de mayor libertad y tranquilidad en un entorno semi-supervisado gracias a la automatización y planificación que permite el prototipo.

### WebApp 13 - Página inicial de presentación

Por último, cabe destacar que, para hacer más ameno el “aterrizaje” en la aplicación, así como para dotar a la página de una mínima funcionalidad para aquellos que no disponen de una cuenta de administrador, presentando el proyecto, se ha creado una página inicial con la información más básica y respondiendo a las que quizá sean las dos preguntas más importantes, el porqué de este proyecto, la causa que nos ha traído hasta este punto y el cómo se ha atajado dicha causa o, cambiando el sujeto, la descripción de esa solución que vertebran tanto este documento como las funcionalidades que se despliegan en la página web.

### Aspectos del diseño

Como se ha podido apreciar en las capturas pertenecientes a la aplicación, se ha optado por un diseño minimalista y de aspecto tecnológico, con un fondo oscuro y letra blanca, conseguimos así, por un lado, centrar la atención del usuario y quedarnos con las necesidades fundamentales, mientras que, por otro lado, facilitamos la legibilidad y descanso visual en aparatos electrónicos que aporta un fondo cercano al color negro.

Los colores son reflejo del planteamiento minimalista, reducimos la gama a lo necesario tomando los colores verde y rojo asignados a conceptos enfrentados, acceso y salida, crear y borrar..., y añadimos dos más, el amarillo, relacionado con conceptos de señalización de peligro y el azul por ser un color asociado a la calma y muy empleados en el ámbito sanitario. Por otro lado, tomaremos colores de apoyo, blanco roto para la letra, gris oscuro en el fondo para rebajar la seriedad y la connotación de luto del color en nuestra cultura y por último un negro aclarado ligeramente para algunos detalles más oscuros.



WebApp 14 - Paleta de colores empleada en la aplicación

La portada se ha estructurado en grandes formatos a modo de dístico, para transmitir apertura y estructura en las ideas, mientras que se ha jugado con el balance de los colores y la apariencia de las letras, con un estilo más *naive*, para transmitir en este primer acercamiento a la aplicación una sensación informal y desenfadada, acentuando la sencillez de uso.

Por último, cabe destacar que en este intento minimalista y de simplificación hasta quedarnos solo con lo más fundamental e indispensable, se ha optado para grandes formatos con una restricción del espacio disponible a la pantalla, evidentemente es imposible poder trabajar con registros sin algún tipo de paginación o slider, pero se ha condensado en una tabla para que todas las herramientas disponibles para cada momento sean visibles a plena vista.

## Sketch de Arduino

### Bibliotecas empleadas

Antes de comenzar a hablar de las características particulares de programa de Arduino, nos detendremos brevemente a comentar acerca de las librerías que vamos a importar y usar en la escritura de este.

De forma directa vamos a emplear:

- **SoftwareSerial:** Esta librería la emplearemos para poder usar como puerto serie con conexión UART los pines conectados al reproductor de música, no es necesaria para nuestro proyecto, pero, si se pretende emplear un Arduino que no tenga más que el puerto serie 0, deberemos incluirla para adicionarla en la declaración del objeto que controla el reproductor.
- **DFRobotDFPlayerMini:** Esta librería es propia del reproductor de música, en su creación precisa de una conexión UART, ya sea por hardware o software, se puede emplear la librería antes mencionada o, en nuestro caso, iniciar el puerto serie y emplearlo.
- **Keypad:** Se trata de una librería muy sencilla para el control de teclado de membra, podríamos haberlo controlado por código, pero con el objetivo de condensar el programa haremos uso de esta.
- **Adafruit\_SSD1306:** Esta librería sirve para el control de la pantalla, precisa de contar con otras librerías de las que hablaremos más adelante y tiene gran capacidad gráfica.
- **RTCLib:** Hablamos de una librería que permite el control del módulo de tiempo real e incluye un nuevo tipo de dato (fechas).
- **Servo:** Es una librería básica que nos va a permitir controlar los servomotores, existen ejemplos propios de Arduino.

De forma indirecta, necesitaremos contar con una serie de librerías puesto que son necesarias para el funcionamiento del resto, estas son:

- **Adafruit\_GFX**
- **Wire**
- **SPI**

## Esquema de uso

En el siguiente esquema, podemos ver una simplificación del funcionamiento del programa que desarrollaremos en el presente apartado.

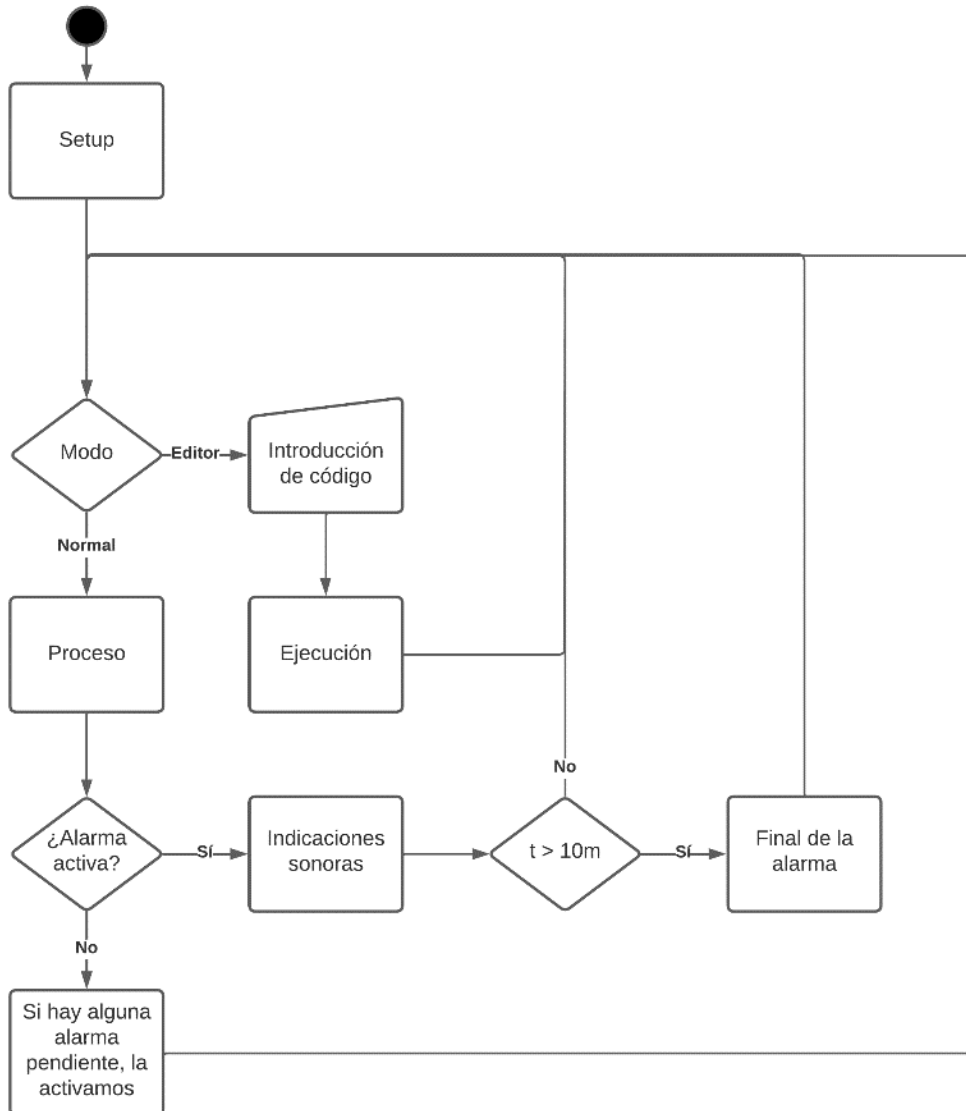
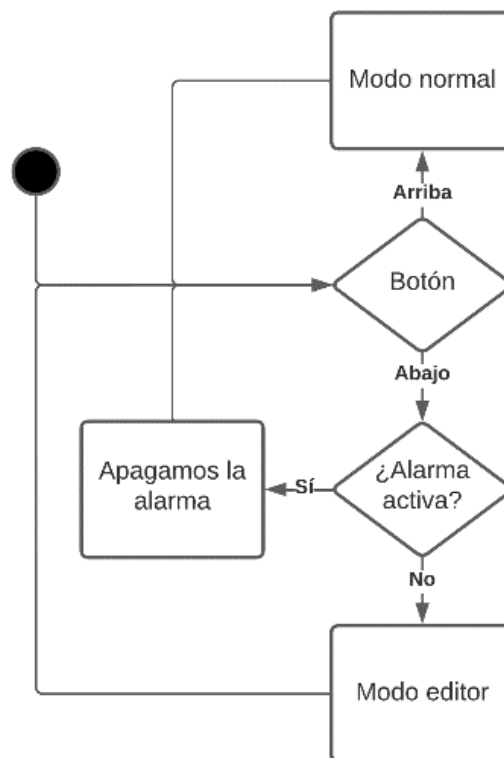


Ilustración 18 - Esquema de uso del prototipo por modos

Cuando el dispositivo se inicia, debe pasar, primero, por un proceso de encendido, donde se declaran y agregan todos los módulos que se van a emplear, así como se inicializan los parámetros base. Si el modo editor está activo, se inicia la lectura de las teclas introducidas por el gestor mediante el teclado de membrana, una vez que se tienen los cuatro caracteres que componen un código, se busca y, si existe, se ejecuta la acción relativa al mismo. Si estamos ante el modo normal, se comprueba que no haya coincidencias

entre la hora y la alarma, si las hubiese, se despliega una alarma que durará 10 minutos. Durante el tiempo de alarma, se reproducirá varias veces un audio de aviso, si el paciente no pulsa el botón no se soltarán los medicamentos y transcurrido el tiempo, la alarma se desconectará por sí sola. Si el paciente pulsa el botón, se abrirá el compartimento indicado y la alarma cesará, una segunda pulsación es requerida para finalizar el proceso. Los compartimentos, además de una alarma, tienen asociado un estado, esto es para saber si aún no se ha iniciado una alarma, si se inició y los medicamentos se tomaron o si se inició y los medicamentos continúan en el compartimento.

Como es normal, debe existir tanto un factor como un funcionamiento que marque tanto el cambio de modo del dispositivo como la lógica aplicada al botón, el siguiente diagrama intenta simplificar estos factores:



*Ilustración 19 - Esquema del cambio en el modo de funcionamiento del prototipo*

Como podemos ver el programa puede ser alterado durante su funcionamiento normal para dar paso al modo editor, esto puede ser realizado siempre y cuando no tengamos ninguna alarma activa, puesto que el botón también se empleará para dar la orden de apertura y cierre de los compartimentos.

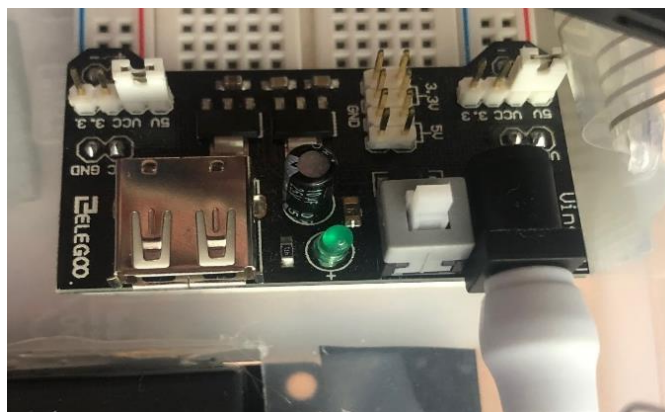
## Aspectos de la construcción del prototipo



*Figura 12 - Prototipo elaborado como TFG*

El prototipo construido es bastante voluminoso, si bien es cierto que un planteamiento más funcional y práctico requeriría disminuir las dimensiones, este gran formato es más versátil en un montaje casero y permite más libertad a la hora de hacer modificaciones en el mismo.

Como podemos ver, se ha elegido un contenedor semitransparente para albergar toda la electrónica y sobre el que disponer diferentes elementos con los que, tanto el cuidador como el paciente, interactúan en el uso normal. El enchufe en la parte trasera es la encargada de alimentar la protoboard y, por ende, los motores que controlan los dispensadores.



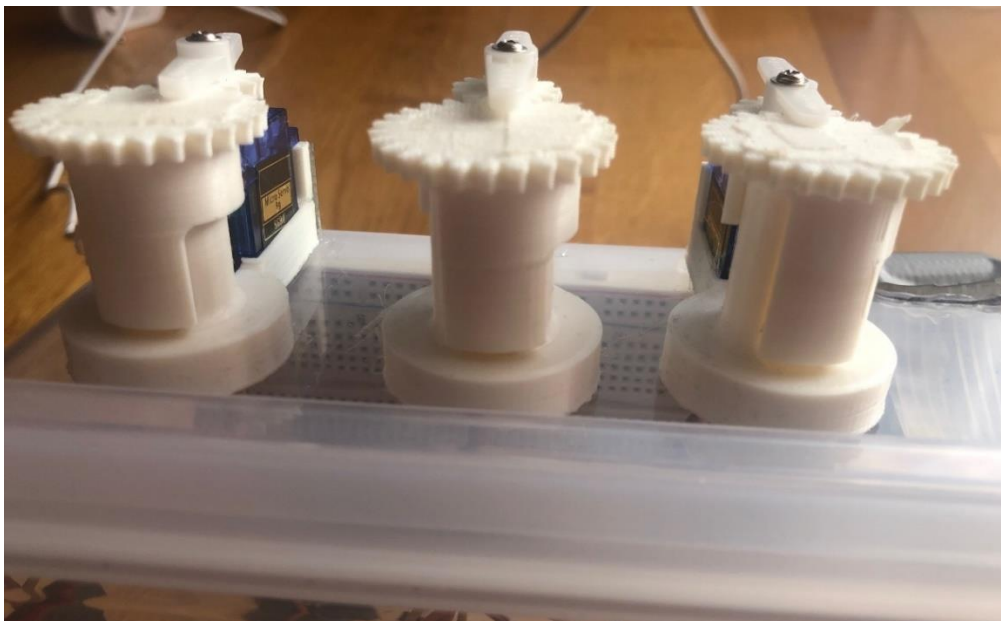
*Figura 13 - Alimentación de la placa de prototipado*





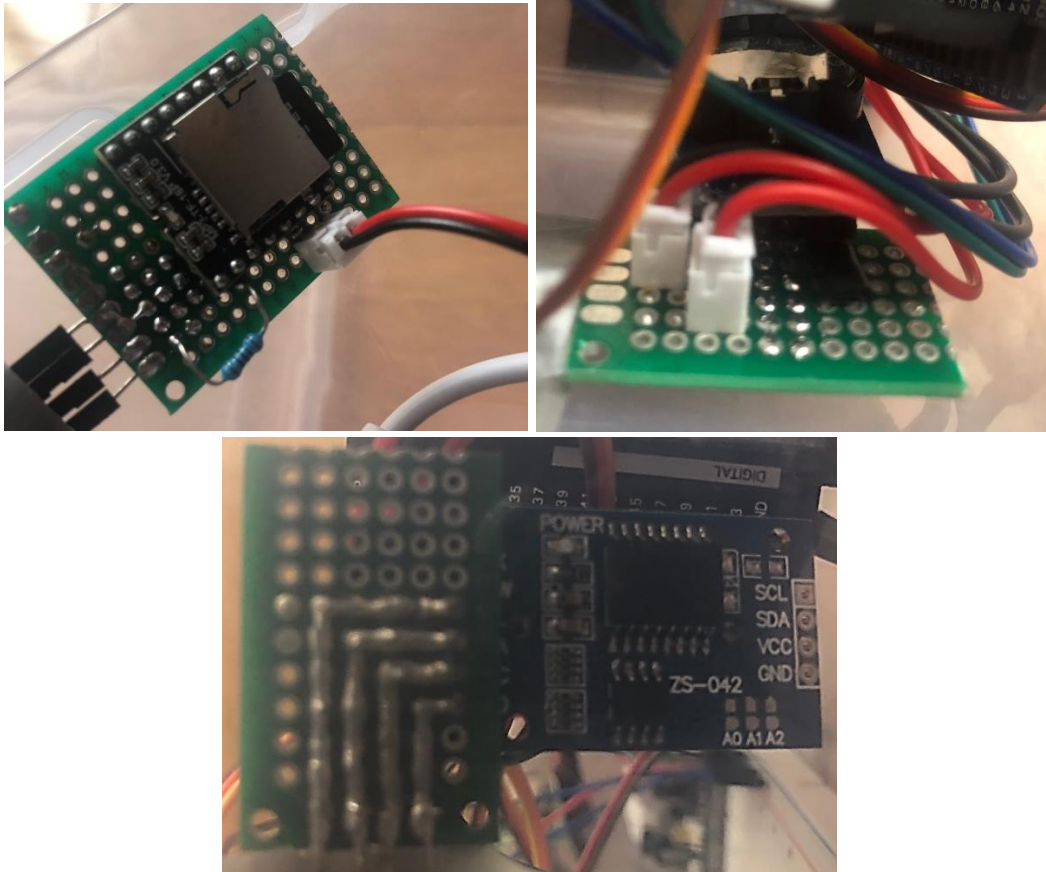
*Figura 14 - Parte superior del pastillero*

La distribución elegida para los componentes externos dispone el display en la parte superior izquierda, junto con un botón para la interacción del paciente, en la parte inferior derecha, se encuentra el teclado de membrana junto al altavoz. En el lateral se encuentran dispuestos los tres motores encajados en los dispensadores del pastillero.



*Figura 15 - Dispensadores acoplados en el pastillero*

En el interior podemos apreciar, además de todo el cableado, una protoboard adherida a la parte superior y el Arduino atornillado en un lateral, diferentes módulos que se han soldado y dispuesto de modo que su inclusión, cambio o retirada sea práctico y orientado al objetivo de la modularización.



*Figura 16 - Detalles de las placas de prototipado soldadas con los componentes modulares*

En estos detalles pueden apreciarse las placas de prototipado incorporadas en el modelo, las cuales han servido para independizar, por un lado, tanto el display como el altavoz, que pueden ser fácilmente retirados y sustituidos gracias a los conectores que vemos en la imagen y por otro el módulo de sonido y el de reloj de tiempo real, los cuales se han aislado del resto de la estructura y solo se ven unidos por cuatro cables en cada caso.

## Conclusiones

### Visión global del proyecto

El prototipo alcanzado se trata de una solución integral que responde a las necesidades tanto del paciente como del cuidador. Con un enfoque interdisciplinar, se ha obtenido una serie de herramientas que abarca el amplio espectro de pasos desde la gestión hasta la toma del medicamento.

Por un lado, se ha trabajado con el lenguaje C++ propio de Arduino en el entorno de desarrollo del mismo para obtener el programa que se ha introducido en el controlador, se ha esquematizado los componentes electrónicos obteniendo el esquema de conjunto mediante el programa Fritzing, soldado a las placas de prototipado y cableado el conjunto de módulos para conformar un prototipo al cual se le han adosado unos dispensadores de creación propia, modelados con Autodesk Inventor, software que también se ha empleado para obtener los archivos STL que, por medio de su transformación a G-code, a través del software Cura, han sido realizados empleado una impresora 3D Anet modelo ET4.

Por otro lado, se ha elaborado una página web con el lenguaje de programación Python bajo el marco de trabajo Django, la cual cuenta con una base de datos creada mediante el programa DB Browser for SQLite. Para dotar de estilo propio a la página se ha empleado el lenguaje CSS partiendo de la base del marco de trabajo Materialize pero personalizándolo para ajustar las diferentes páginas programadas con HTML y a las que se les ha dado funcionalidad por medio de JavaScript.

Todo esto con un enfoque pragmático y filosofía minimalista, reduciendo los elementos a la mínima expresión útil en aras de conseguir tanto una simplificación de los procesos.

### Consecución de objetivos

Para responder a este apartado iremos revisando todos los puntos expuestos al inicio del documento y comentando las acciones llevadas a cabo.

1. Interacción con el usuario a nivel Hardware con el dispositivo simple y minimalista.

Se ha conseguido reducir la interacción del paciente a un botón y el control administrativo del dispositivo mediante códigos cortos.

2. Interacción con el usuario a nivel software simple pero completa.

La base de datos construida ha seguido la filosofía minimalista del conjunto, se han reducido el número de elementos en pantalla, acotado el campo visual para mostrar todos los elementos con los que se puede interactuar y empleado dos recursos como son la posición y los colores de manera intuitiva y auto explicativa.

3. Restricción de acceso a los medicamentos.

Los compartimentos mantienen seguros los medicamentos hasta que estos son dispensados y sería necesario el uso de la violencia para forzarlos.

4. Avisos suficientemente evidentes.

El audio de alarma, el cual puede ser modificado si resulta demasiado estridente y disruptivo, constituye una señal acústica evidente y diferenciable.

5. Comunicación visual básica mediante Hardware del estado del dispositivo.

La inclusión de la pantalla nos granjea la compleción de este punto.

6. Diseño modular para maximizar la adaptabilidad a necesidades no contempladas.

No solo los dispensadores son unitarios e independientes, si nos acercamos al apartado sobre el montaje podremos ver reflejada esta filosofía en la construcción.

7. Capacidad de recolección de datos y su transmisión a una persona responsable.

La pestaña de mensajes es capaz de almacenar los datos relativos a las tomas del paciente y si estas han sido efectuadas.

8. Capacidad de comunicación del proceso de una forma profesional de modo que la persona sea conocedora de la información relativa a la toma del medicamento.

Este punto ha sido planteado de forma patente con la inclusión de audios por defecto y la permisión de audios personalizados, pero puede ser modificado pues priman los intereses del paciente a discreción del cuidador.

## Aspectos de mejora

### Cambio de motor.

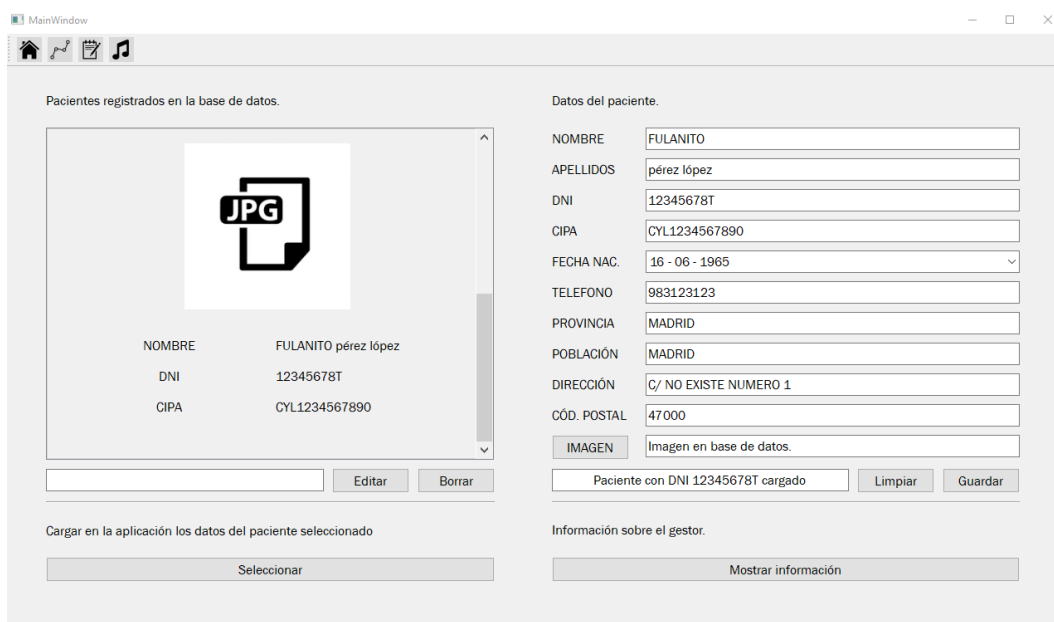
- Debido tanto a que el dispositivo en sí es una prueba de capacidades como a que el precio de su elaboración ha debido ajustarse a las posibilidades, el motor elegido está sobredimensionado. Existen opciones más caras pero que permiten la reducción del espacio necesario para cada celda, existen soluciones comerciales como los “hermanos menores” del servo empleado, los de 5 y 2 gramos. El modelo podría adaptarse e incluso introducir partes mecanizadas como los rodamientos que permitirían reducir la fricción y emplear motores menos potentes.
- Otra opción es el empleo de motores piezoeléctricos, todavía son una tecnología poco empleada, pero permiten realizar pequeños movimientos con elementos de muy reducido tamaño, el problema, una vez más es el precio.
- Por último, otra opción que se ha estudiado es el empleo de actuadores hasel. La implementación de robótica blanda de bajo costo parece una solución inteligente, el problema es que no es un producto comercial o que se pueda conseguir de forma sencilla, y el plantear una pieza que se adecue constituiría por sí mismo un trabajo completo.

### Dispensador de cinta transportadora.

- El dispensador ejecutado no es ni la primera idea ni la mejor, pero si el resumen de multitud de intentos hasta dar con un elemento fabricable según mis capacidades. Un mejor dispensador podría estar compuesto por dos cintas transportadoras agujereadas y enfrentadas, de modo que las pastillas quedarían atrapadas en su interior y el discurrir de ambas dos cintas soltarían os medicamentos. La versatilidad de este dispensador se explica si nos imaginamos lo fácil que sería automatizar un proceso en el que, incluso, podríamos introducir visión artificial para la inspección, por no hablar de lo sencillo que sería mover dos cintas del mismo tamaño con un solo motor pudiendo albergar gran cantidad de unidades del medicamento.

App que se conecte a la base de datos.

- La primera opción fue la de implementar una aplicación de escritorio la cual fue parcialmente construida como se puede apreciar en las capturas que se adjuntan, el problema radica en dos vertientes. La primera es lo limitado de la aplicación por haber sido implementadas con un software que a nivel gráfico ha quedado desfasado. La segunda es la falta de protección de los datos que ofrecía la aplicación y que con mis conocimientos actuales no podía subsanar



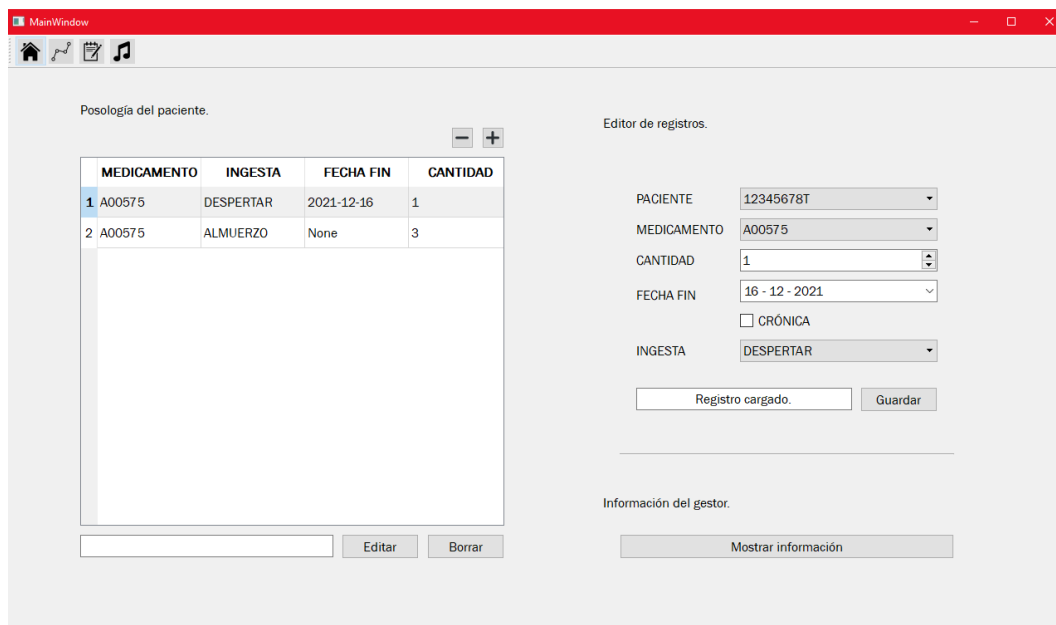


Figura 17 - Capturas de la aplicación de escritorio que se empezó a elaborar y se migró

- La idea me sigue pareciendo interesante y pienso que tratar la aplicación web como una página de administración descentralizando alguna de sus funciones a una aplicación de escritorio que pudiera conectarse con la base de datos sería una muy buena opción.

#### Conectividad Arduino-app.

- En relación con el punto anterior, la aplicación web tiene una grave falla de la que carecía la aplicación de escritorio, no puede conectarse al Arduino de forma directa. Empleando Python, somos capaces de conectarnos al puerto serie de la placa mediante el propio script e incluso programar la placa, son múltiples los ejemplos que podemos encontrar en internet de proyectos que implementan esta característica y considero que constituiría un grado más de automatización al proceso, eximiendo al responsable del cargo del paciente de la necesidad de tener mayores conocimientos sobre lo que es Arduino y como se emplea un sketch.

#### Compatibilidad con soluciones existentes.

- En el presente documento hemos hablado de soluciones existentes como la de (Ti-Medi, s.f.), otro enfoque del pastillero podría ser el de tomar como punto de partida soluciones como la descrita las cuales atajan el problema farmacéutico, pero no el humano.

Librería propia para el control del display.

- Actualmente empleamos una librería para el control de la pantalla de Adafruit, una solución muy testeada y potente pero que podría ser sustituida por una librería propia de modo que se facilitase la forma en la que se muestra el texto por pantalla, se podrían incluir mensajes para el paciente o adicionar gráficos propios de forma más compacta.

Circuito impreso.

- Este punto de mejora es obvio, para un prototipo el empleo de una protoboard y cableado es admisible, pero un producto serio, compacto y estable ha de tener un circuito impreso.

Alimentación integrada.

- Una de las mayores limitaciones que incluye el prototipo es la alimentación, he intentado atajar el problema de la mejor manera posible, pero la alimentación a 3.5, 5 y 9 Voltios es un problema de gestión que habría que afrontar en un futuro

Control de motores mediante lógica.

- Con la incorporación de demultiplexores, podemos aumentar la capacidad de motores de los que disponemos en el dispositivo disminuyendo el número de salidas necesarias, es un cambio para plantear seriamente.

Memoria eeprom.

- Aunque el Arduino dispone de su propia memoria no volátil la cual puede ser empleada, esta tiene una vida útil relativamente corta, por lo que incluir una memoria para almacenar las horas y días de las tomas podría ser una buena mejora.

Módulo sintetizador.

- Otra de las posibles mejoras que se me han ocurrido ha sido la posibilidad de sustituir el reproductor de música por un módulo sintetizador capaz de reproducir textos de forma que se aumentase el grado de interacción con el paciente



## Bibliografía

- Al-Shammary, R., Mousa, D., & Esmaeili, S. (9 de 2018). The Design of a Smart Medicine Box. *26th Iranian Conference on Electrical Engineering, ICEE 2018*, 130-134. doi:10.1109/ICEE.2018.8472586
- Antoun, W., Abdo, A., Al-Yaman, S., Kassem, A., Hamad, M., & El-Mou Cary, C. (7 de 2018). Smart Medicine Dispenser (SMD). *Middle East Conference on Biomedical Engineering, MECBME, 2018-March*, 20-23. doi:10.1109/MECBME.2018.8402399
- Hayes, T., Hunt, J., Adami, A., & Kaye, J. (2006). An electronic pillbox for continuous monitoring of medication adherence. *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, 6400-6403. doi:10.1109/IEMBS.2006.260367
- Jabeena, A., Sahu, A., Roy, R., & Basha, N. (6 de 2018). Automatic pill reminder for easy supervision. *Proceedings of the International Conference on Intelligent Sustainable Systems, ICISS 2017*, 630-637. doi:10.1109/ISS1.2017.8389315
- Kanhasinwattana, J., Yawila, N., Tithada, T., & Kamyod, C. (3 de 2020). Smart Pill Box System for Bipolar Disorder Patients. *2020 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering, ECTI DAMT and NCON 2020*, 54-57. doi:10.1109/ECTIDAMTNCN48261.2020.9090716
- Ministerio de Sanidad, Consumo y Bienestar Social. (2019). *Plan Integral de Alzheimer y otras Demencias (2019-2023)*. GOBIERNO DE ESPAÑA, MINISTERIO DE SANIDAD, CONSUMO Y BIENESTAR SOCIAL. Madrid: CENTRO DE PUBLICACIONES DEL MINISTERIO DE SANIDAD, CONSUMO Y BIENESTAR SOCIAL. doi:(NIPO) 731-19-108-2
- Othman, N., & Ek, O. (12 de 2016). Pill dispenser with alarm via smart phone notification. *2016 IEEE 5th Global Conference on Consumer Electronics, GCCE 2016*. doi:10.1109/GCCE.2016.7800399
- Parra, J., Valdez, W., Guevara, A., Cedillo, P., & Ortiz-Segarra, J. (4 de 2017). Intelligent pillbox: Automatic and programmable Assistive Technology device. *Proceedings of the 13th IASTED International Conference on Biomedical Engineering, BioMed 2017*, 74-81. doi:10.2316/P.2017.852-051
- Rajan, B., Brightny, A., Rebeak, C., Poojasen, S., Vijay, P., Suresh, M., & Rohith Alex, R. (2021). Smart Pill Box with Reminder to Consume and Auto-Filling Process Using IOT. *Proceedings of the 5th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), I-SMAC 2021*, 40-44. doi:10.1109/I-SMAC52330.2021.9641043

- Rosdi, W., Suhaimi, S., Lazam, N., Alias, A., Abdullah, F., & Azemi, S. (2021). Smart Pill Dispenser with Monitoring System. *IEEE Symposium on Wireless Technology and Applications, ISWTA, 2021-August*, 58-62. doi:10.1109/ISWTA52208.2021.9587450
- Ti-Medi. (s.f.). *Sistema Personalizado de Dosificación*. Recuperado el 12 de 3 de 2022, de Robot SPD: <https://ti-medi.com/>
- Tsai, H., Tseng, C., Wang, L., & Juang, F. (7 de 2017). Bidirectional smart pill box monitored through internet and receiving reminding message from remote relatives. *2017 IEEE International Conference on Consumer Electronics - Taiwan, ICCE-TW 2017*, 393-394. doi:10.1109/ICCE-CHINA.2017.7991161
- World Health Organization. (21 de Septiembre de 2020). *Organización Mundial de la salud: Demencia*. Recuperado el 28 de Septiembre de 2021, de Demencia: <https://www.who.int/es/news-room/fact-sheets/detail/dementia>