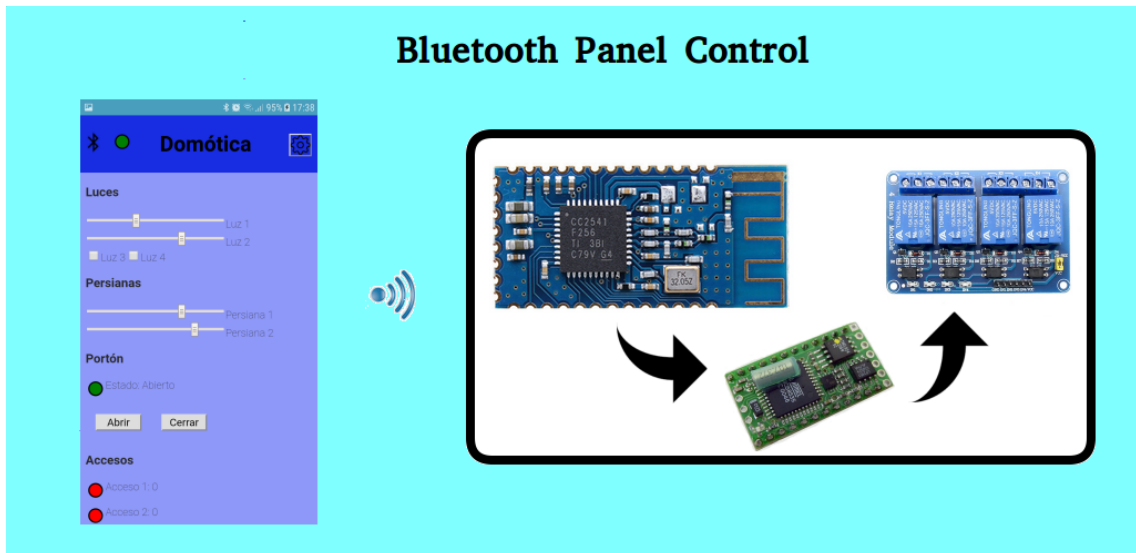


# Aplicación “BluApp” (panel de control bluetooth)



## Índice

Introducción - Aplicación “panel de control bluetooth” (BluApp).....	3
Funcionamiento de la aplicación BluApp como consola de TX/RX.....	4
Funcionamiento de la aplicación BluApp como panel de control.....	5
Formato detallado de comando modificar nombre (00).....	7
Formato detallado de comando agregar botón (01).....	8
Formato detallado de comando agregar caja de texto (02).....	9
Formato detallado de comando agregar caja de texto con múltiples contenidos (0A).....	9
Formato detallado de comando agregar visor (03).....	11
Formato detallado de comando agregar objeto html (04).....	12
Formato detallado de comando almacenar aplicación (05).....	12
Formato detallado de comando borrar aplicación (06).....	12
Lista completa de mensajes.....	13
Editar la aplicación BluApp localmente.....	14
Ejemplo de configuración para un proyecto de domótica.....	14
Contacto.....	17

## **Introducción – Aplicación “panel de control bluetooth” (BluApp)**

Actualmente los smartphones ofrecen al usuario capacidades avanzadas de interfaz, sensorización y conectividad. ¿Pero qué sucede si el usuario desea establecer una conexión entre su smartphone y un sistema electrónico diseñado por él? En esta situación el usuario se encuentra con dos dificultades:

1. Simultáneamente con el diseño del sistema electrónico es necesario diseñar la aplicación del smartphone.
2. El diseño del sistema electrónico debe disponer de alguna de las conectividades avanzadas del smartphone.

La mayor complicación se encuentra en el primer punto, donde se agrega una complejidad adicional en el diseño de la solución general que es realizar el programa completo en el smartphone. Por ejemplo si el usuario está pensando en utilizar módulos Arduino en su sistema electrónico, además de poder desenvolverse en dicho entorno debería conocer y desarrollar una aplicación en su smartphone.

La aplicación “panel de control bluetooth” (de aquí en más BluApp) permite diseñar un panel de control en su dispositivo android de su proyecto electrónico. Para ello, utiliza el módulo HC-05 o HM-10 y permite dos posibilidades, usar como terminal serie para depurar la interfaz de comunicación con un microcontrolador, o desarrollar un panel de control personal según las necesidades de nuestro proyecto.

Para configurar el panel de control se puede utilizar la misma interfaz de comunicación bluetooth desde el microcontrolador o editar desde la misma aplicación BluApp. En ambos casos es posible almacenar el panel de control de forma permanente, es decir sin necesidad de ser cargado cada vez que se inicie la aplicación. En los comandos de comunicación para editar el panel de control se utiliza lenguaje html, esto posibilita una extensa gama de posibilidades de ajustes personales (color de botones, agregar texto, tamaños, etc.). Los comandos permiten de forma simple y rápida agregar botones de control de comandos, lecturas de variables remotas, etc.

La aplicación BluApp viene con un ejemplo de configuración de panel de control.

Para quien es la aplicación BluApp:

- Ideal para aprender electrónica de un modo divertido.
- Ideal para prototipado rápido de una nueva idea.

- Ideal para exhibir su proyecto.
- Ideal para controlar su producto final

El usuario solamente debe utilizar un módulo bluetooth HM10 o HC05 del lado del sistema electrónico el cual requiere una configuración mínima.

Nota: No está disponible ni entrenamiento ni soporte de la aplicación BluApp. El uso de la aplicación BluApp y los ejemplos brindados son bajo su propio riesgo.

## Funcionamiento de la aplicación BluApp como consola de TX/RX

La pantalla principal de la aplicación BluApp se puede observar en la figura 1.



Figura 1

La misma consta de un encabezado con un indicador de estado de conexión (rojo desconectado y verde conectado) y un botón para entrar en las configuraciones. En el cuerpo principal se observa un cuadro con el historial de las comunicaciones entre la aplicación BluApp y el módulo bluetooth y a continuación un cuadro de texto modificable y un botón de envío.

Si es pulsado el botón de configuración, luego se despliega el sub menú “Conexión” y se pulsa el botón “Actualizar Bluetooth Classic” (o “Actualizar Bluetooth LE”), es posible ver la lista de dispositivos que se encuentran disponibles para conectarse. Luego se debe pulsar en la lista el dispositivo a conectar.

En la figura 2 se puede observar una conexión establecida con el dispositivo “mcm”, en el encabezado el indicador es de color verde (con conexión a dispositivo) y por último se agrega la posibilidad de desconectarse por medio del botón “Desconectar”.

Pulsando nuevamente el botón de configuración es posible volver a la pantalla principal. En este momento es posible observar los caracteres recibidos remotamente (por medio del cuadro de historial) y los que son transmitidos desde la aplicación BluApp (por medio del botón de “Enviar”).

Utilizando la aplicación BluApp de esta forma es posible comenzar a depurar la comunicación con el sistema electrónico que estamos desarrollando, por ejemplo determinar si efectivamente se está transmitiendo algo, si el orden de transmisión de caracteres es el deseado, etc.

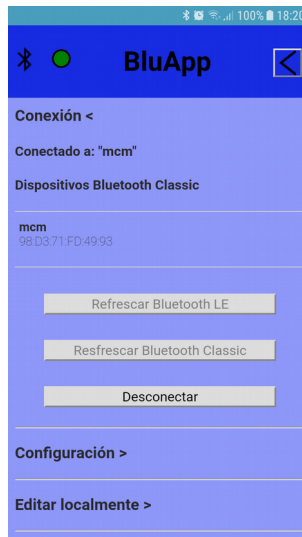


Figura 2

## Funcionamiento de la aplicación BluApp como panel de control

Una vez que existe una comunicación establecida entre la aplicación BluApp y el sistema electrónico y somos capaces de enviar y recibir las tramas de caracteres que deseamos es posible personalizar la aplicación BluApp según nuestras necesidades particulares.



Figura 3



Figura 4

Si no queremos ver el cuadro de historial y la transmisión de test, debemos acceder a los ajustes, luego desplegar el sub menú "Configuración" y deseleccionar las opciones "Mostrar el historial de comunicación" y "Agregar transmisión de test" (ver figura 3). Al volver a la pantalla principal se

observa que la misma está vacía y lista para ser adaptada a nuestro proyecto (ver figura 4).

En el sub menú “Configuración” también es posible especificar si deseamos que la aplicación transmita un carácter “\n” al final de cada transmisión y/o luego de cada carácter “]”. Por último es posible indicar que queremos visualizar el historial en formato hexadecimal, en este modo la transmisión de test se debe editar con dos dígitos hexadecimales (que representarán el código ascii) seguidos de un espacio, por ejemplo la cadena hexadecimal “30 31 32 32 0A” se refiere a la cadena “0123\n”.

Utilizando la aplicación BluApp como panel de control se pueden agregar los siguientes objetos:

- Botón que envía una cadena de texto configurable
- Caja de texto editable
- Agregar caja de texto con múltiples contenidos
- Botón asociado a un cuadro de texto (se envía contenido del mismo)
- Visor de variable remota con indicador de validez
- Título personalizado
- Objetos html

Para agregar nuevos objetos en la pantalla principal de la aplicación BluApp se deben enviar comandos que posean el siguiente formato:

```
[NN;aaa;...;xxxxxx]\n
```

Donde:

- Los caracteres “[” y “]” indican el inicio y fin de la trama de comando en cuestión
- La trama de comandos se divide en distintos campos separados por el carácter “;”
- Los caracteres “NN” indican el número de comando
- Los caracteres “aaa”, “xxxxxx”, etc. representan los parámetros de la trama en cuestión. Estas cadenas de caracteres tienen longitud variable
- El caracter “\n” indica la terminación del comando

En el caso que uno de los campos sea código html, es importante tener en cuenta que todos los identificadores a asignar deben comenzar con “\_id\_”. En caso contrario el comando será descartado y la aplicación BluApp devolverá un mensaje del tipo “[0B;error00]\n”.

Por ejemplo, para modificar el título de la aplicación BluApp con el nombre “Personal” es necesario enviar a la aplicación BluApp la siguiente trama: “[00Personal]\n”. Es decir que el número de comando para este caso es el “00”.

Además de agregar distintos objetos a la aplicación BluApp por medio de comandos, es posible indicarle que guarde permanentemente esa configuración o la borre. Por lo tanto el sistema electrónico no necesariamente tiene que configurar el panel de control cada vez que inicie y por otro lado es posible que borre la configuración actual y la actualice de forma dinámica.

En la tabla 1 se pueden observar la lista de los comandos posibles para agregar objetos a la aplicación BluApp.

<b>NN</b>	<b>Nombre de Comando</b>
00	Modificar nombre de la aplicación
01	Agregar botón
02	Agregar caja de texto
0A	Agregar caja de texto con múltiples contenidos
03	Agregar visor
04	Agregar objeto html
05	Almacenar permanentemente la aplicación personalizada
06	Borrar la aplicación personalizada

Tabla 1

## **Formato detallado de comando modificar nombre (00)**

El formato de este comando es el siguiente:

```
[00Personal]\n
```

En este caso “Personal” es el nombre que se mostrará en el encabezado de la aplicación BluApp.

## **Formato detallado de comando agregar botón (01)**

El formato de este comando es el siguiente:

```
[01;_id_xx;yyyy;zzzz;<button id='_id_xx'>Boton 1</button>]\n
```

Se puede observar que este comando posee cinco campos. El primer campo es el número de comando (en este caso “01”).

El segundo campo indica el nombre de identificador que será utilizado en la aplicación BluApp para este objeto, en este caso se utilizará el identificador “\_id\_xx”. Es importante recordar que todos los identificadores que se utilicen deben comenzar con “\_id\_”.

El tercer y cuarto campo están relacionados de la siguiente forma, si deseamos que el botón envíe una cadena fija cada vez que es pulsado el campo tres especifica cual es esta cadena fija y el campo cuatro debe ser “undefined”. Por ejemplo, si deseamos que se envíe la cadena fija “pepe” cada vez que se pulse el botón el comando debe ser:

```
[01;_id_xx;pepe;undefined;<button id='_id_xx'>Boton 1</button>]\n
```

En este ejemplo cada vez que se pulse el botón la aplicación BluApp enviará al dispositivo remoto el siguiente mensaje:

```
[07pepe]\n
```

En cambio si queremos asociar este botón al contenido de una caja de texto el campo tres debe ser “undefined” y el campo cuatro debe ser el identificador de la caja de texto, por ejemplo:

```
[01;_id_xx;undefined;_id_caja;<button id='_id_xx'>Boton 1</button>]\n
```

En este ejemplo cada vez que se pulse el botón la aplicación BluApp enviará al dispositivo remoto el siguiente mensaje:

```
[08;_id_caja;ejemplo]\n
```

En este caso se utilizó previamente un comando 02 para agregar una caja de texto y el usuario editó en la misma el texto “ejemplo”.

Por último, el quinto campo es el código html que agregará el objeto botón en la aplicación BluApp. Notar que el id utilizado en este campo es el mismo que contiene el segundo campo.



## Formato detallado de comando agregar caja de texto (02)

Este comando permite agregar una caja de texto en la aplicación BluApp que permita editar texto y luego enviarlo remotamente al presionar un botón configurado a tal fin. El formato de este comando es el siguiente:

```
[02;_id_ww;<input type='text' id='_id_ww'>Texto 1</input>]\n
```

Se puede observar que este comando posee tres campos. El primer campo es el número de comando (en este caso "02").

El segundo campo indica el nombre de identificador que será utilizado en la aplicación BluApp para este objeto, en este caso se utilizará el identificador "\_id\_ww". Es importante recordar que todos los identificadores que se utilicen deben comenzar con "\_id\_" y que no deben repetirse entre objetos distintos.

Por último, el tercer campo es el código html que agregará el objeto caja de texto en la aplicación BluApp. Notar que el id utilizado en este campo es el mismo que contiene el segundo campo. En el código html de este campo es posible indicar tres tipos distintos de entradas:

- 1) type='text' : para indicar una entrada caja de texto
- 2) type='range' min='-10' max='10': para indicar una entrada de barra deslizable con límite inferior de -10 y límite superior de 10
- 3) type='checkbox': para indicar una entrada checkbox

## Formato detallado de comando agregar caja de texto con múltiples contenidos (0A)

Este comando permite agregar una caja de texto (que puede ser visible o no en la aplicación BluApp) cuyo contenido esta determinado por otras fuentes (por ejemplo: input type='checkbox', input type='range', etc.).

El contenido de esta caja se puede enviar ante un pedido en la aplicación BluApp (asociándolo a un botón por medio del comando "01") o enviarlo periódicamente cada un tiempo especificado en mSeg.

El formato de este comando es el siguiente:

```
[0A;_id_ee;xxx;_id_1;_id_2;_id_3;<input type='text' id='_id_ee'>Texto 1</input>]\n
```

Este comando posee un número variable de campos. La cantidad de campos mínima es cuatro. Los campos obligatorios son el primer campo (“0A” tipo de comando), el segundo campo (en el ejemplo “\_id\_ee” que es el identificador asignado al objeto html del comando), el tercer campo (en el ejemplo “xxxx” que es la periodicidad de la respuesta expresada en mSeg) y el último campo que es el código html del objeto agregado.

A medida que se irán asociando objetos es necesario ir agregando su identificador luego del tercer campo (en el ejemplo son “\_id\_1”, “\_id\_2” e “\_id\_3”).

Por ejemplo, en el caso que el campo “xxxx” sea “100” y que “\_id\_1”, “\_id\_2” y “\_id\_3” son asociados a cajas de texto conteniendo las cadenas “Texto1”, “Texto2” y “Texto3” respectivamente, entonces la aplicación BluApp enviará periódicamente cada 100 mSeg la trama:

```
[08;_id_ee;_id_1;Texto1;_id_2;Texto2;_id_3;Texto3]\n
```

Independientemente de la periodicidad, en el momento que se modifica el valor de alguna de las entradas esa modificación es informada instantáneamente. Si el contenido de la entrada con identificador “\_id\_2” pasa de tener un valor “Texto2” a uno “Texto2modificado”, la aplicación BluApp enviará una respuesta instantánea como la siguiente:

```
[08;_id_ee;_id_2;Texto2modificado]\n
```

Se puede observar que en el mensaje anterior solamente se hizo referencia al objeto “\_id\_2”, pero en la siguiente respuesta periódica se tendrá:

```
[08;_id_ee;_id_1;Texto1;_id_2;Texto2modificado;_id_3;Texto3]\n
```

Si no es necesario disponer de las entradas periódicamente y solamente necesitamos tener la información de los cambios que se van produciendo, el tercer campo del comando “0A” (periodicidad) debe ser configurado con un valor “0”, es decir sin periodicidad.

En el ejemplo visto se configura una entrada con identificador “\_id\_ee”, esta entrada debe ser pensada como un contenedor de las otras entradas definidas (en este caso “\_id\_1”, “\_id\_2” e “\_id\_3”), es decir que el contenido de “\_id\_ee” será una cadena de caracteres concatenada con los valores de los otros identificadores. En nuestro caso inicialmente el valor de la entrada “\_id\_ee” era de “;\_id\_1;Texto1;\_id\_2;Texto2;\_id\_3;Texto3” y luego paso a ser “;\_id\_1;Texto1;\_id\_2;Texto2modificado;\_id\_3;Texto3”.

### Formato detallado de comando agregar visor (03)

Este comando permite visualizar una variable remota en la aplicación BluApp, indicando además si es actual u obsoleta. El formato de este comando es el siguiente:

```
[03;Visor1: ;_id_pp;2000;<div id='_id_pp'></div>]\n
```

Se puede observar que este comando posee cinco campos. El primer campo es el número de comando (en este caso "03").

El segundo campo es el nombre que se mostrará en la aplicación BluApp para este visor. En el caso anterior el nombre sería "Visor1: ".

El tercer campo indica el nombre de identificador que será utilizado en la aplicación BluApp para este objeto, en este caso se utilizará el identificador "\_id\_pp". Es importante recordar que todos los identificadores que se utilicen deben comenzar con "\_id\_" y que no deben repetirse entre objetos distintos.

El cuarto campo indica el tiempo máximo a partir del cual se considera que el valor mostrado de la variable es obsoleta. La unidad de este tiempo es mSeg, en el ejemplo anterior el tiempo es de 2000 mSeg. Para mostrar un valor el sistema electrónico debe enviar el siguiente mensaje de actualización:

```
[09;_id_pp;1234]\n
```

En este caso si el mensaje anterior se envía periódicamente cada un tiempo menor a 2000 mSeg entonces el visor se considerará válido y su indicador será de color verde (ver figura 6), en cambio en el momento en que pase un tiempo mayor a 2000 mSeg sin recibir un mensaje de actualización el visor se considerará obsoleto y su indicador será de color rojo (ver figura 5).

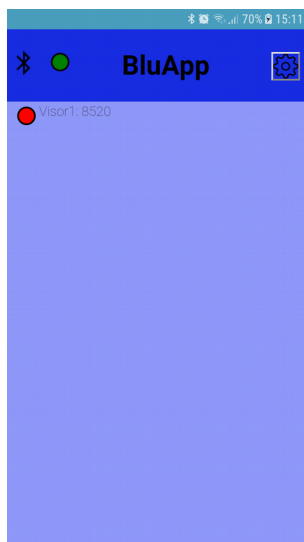


Figura 5

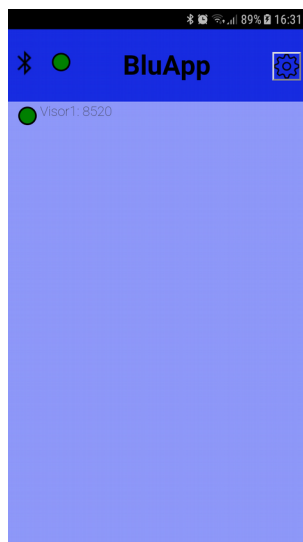


Figura 6

Por último, el quinto campo es el código html que se agregará en la aplicación BluApp. Notar que el id utilizado en este campo es el mismo que contiene el segundo campo.

### **Formato detallado de comando agregar objeto html (04)**

Este comando es útil para agregar objetos html a la aplicación BluApp. El formato de este comando es el siguiente:

```
[04<div>Test</div>]\n
```

En este comando se indica el número de comando (04) y luego el código html a ser agregado en la aplicación BluApp.

### **Formato detallado de comando almacenar aplicación (05)**

Este comando se utiliza cuando es necesario guardar permanentemente todos los objetos agregados en la aplicación BluApp. El formato es el siguiente:

```
[05]\n
```

Esta acción también se puede llevar a cabo entrando en la configuración de la aplicación BluApp, expandiendo el sub menú “Guardar” y luego pulsar el botón “Guardar”.

### **Formato detallado de comando borrar aplicación (06)**

Este comando se utiliza cuando es necesario borrar todos los objetos agregados en la aplicación BluApp y dejarla en blanco. El formato es el siguiente:

```
[06]\n
```

Esta acción también se puede llevar a cabo entrando en la configuración de la aplicación BluApp, expandiendo el sub menú “Guardar” y luego pulsar el botón “Borrar”.

## Lista completa de mensajes

Como se pudo ver además de los mensajes para configurar la aplicación BluApp existen otros mensajes enviados que indican que se pulsó un botón o mensajes para actualizar valores de un visor (mensajes “07” al “09”). En la tabla 2 se puede observar la lista completa de comandos y mensajes.

N°	Ejemplo	Descripción
00	[00Personal]\n	Comando enviado desde el sistema electrónico a la aplicación BluApp para modificar el título
01	[01;_id_xx;yyyy;zzzz;<button id='_id_xx'>Boton 1</button>]\n	Comando enviado desde el sistema electrónico a la aplicación BluApp para agregar un botón
02	[02;_id_ww;<input type='text' id='_id_ww'>Texto 1</input>]\n	Comando enviado desde el sistema electrónico a la aplicación BluApp para agregar una caja de texto
03	[03;Visor1: ;_id_pp;2000;<div id='_id_pp'></div>]\n	Comando enviado desde el sistema electrónico a la aplicación BluApp para agregar un visor
04	[04<div>Test</div>]\n	Comando enviado desde el sistema electrónico a la aplicación BluApp para agregar código html
05	[05]\n	Comando enviado desde el sistema electrónico a la aplicación BluApp para almacenar permanentemente los objetos configurados
06	[06]\n	Comando enviado desde el sistema electrónico a la aplicación BluApp para borrar los objetos configurados
07	[07pepe]\n	Mensaje fijo enviado desde la aplicación BluApp al sistema electrónico cada vez que se pulsa un botón configurado para enviar una cadena de texto fija
08	[08;_id_gg;ejemplo]\n	Mensaje enviado desde la aplicación BluApp al sistema electrónico cada vez que se pulsa un botón configurado para enviar una cadena de texto editada en una caja de texto. Otra posibilidad es que sea un mensaje enviado desde la aplicación BluApp por estar asociado con una caja de texto con múltiples contenidos
09	[09;_id_pp;1234]\n	Mensaje enviado desde el sistema electrónico a la aplicación BluApp para actualizar el valor de un visor
0A	[0A;_id_ee;xxxx;_id_1;_id_2;_id_3;<input type='text' id='_id_qq'>Texto 1</input>]\n	Comando enviado desde el sistema electrónico a la aplicación BluApp para agregar una caja de texto con múltiples contenidos
0B	[0Bxxxx]\n	Mensaje enviado desde la aplicación BluApp al sistema electrónico para indicar error.

Tabla 2

## Editar la aplicación BluApp localmente

Como se pudo ver el agregado de los objetos que permiten personalizar la aplicación BluApp según las necesidades de nuestro proyecto es por medio de la recepción de mensajes.

Para facilitar las cosas la aplicación BluApp permite editar un buffer de texto que contendrá los comandos de configuración y luego enviarlos a su buffer de recepción, de esta forma es posible tener una aplicación personalizada y enfocar el desarrollo del sistema electrónico directamente en los mensajes del tipo "07", "08" y "09".

Al entrar en la configuración de la aplicación BluApp y expandiendo el sub menú "Editar localmente" se puede acceder a una caja de texto editable, seis botones para agregar plantillas de comandos, un botón para limpiar la caja de texto y por último un botón para enviar al buffer de recepción a la aplicación BluApp (ver figura 7).



Figura 7

Para poder utilizar esta característica la aplicación BluApp no debe estar conectada a ningún dispositivo.

## Ejemplo de configuración para un proyecto de domótica

Se dispone de un sistema electrónico con microcontrolador para uso en domótica que posee las siguientes capacidades:

- Salida proporcional Luz 1

- Salida proporcional Luz 2
- Salida on/off Luz 3
- Salida on/off Luz 4
- Salida digital abrir Portón
- Salida digital cerrar Portón
- Entrada estado de portón
- Salida proporcional persiana 1
- Salida proporcional persiana 2
- Entrada sensor acceso 1
- Entrada sensor acceso 2
- Una interfaz Bluetooth por medio de un dispositivo HC-05

El sistema electrónico resuelve las etapas electrónicas de adaptación, control de cargas y otras, pero desarrollar una interfaz o panel de control moderna que permita el control y monitoreo del mismo no es simple. Por este motivo se recurre al uso de la aplicación BluApp.

En primer lugar se debería pensar un borrador de nuestra aplicación, una opción puede ser dividir la pantalla en cuatro áreas separadas por títulos (“Luces”, “Persianas”, “Portón” y “Accesos”). Luego debemos pensar que objetos html necesitamos y que identificador tendrá cada uno (ver Tabla 3).

<b>Código html del comando</b>	<b>Id</b>	<b>Descripción</b>
<h3>Luces</h3>		Título de la primer área “Luces”. Este control se agrega por medio del comando “04”.
<input type=’range’ ..>	_id_1	Barra deslizable para controlar la salida proporcional “Luz 1”. Este control se agrega por medio del comando “04”.
<input type=’range’ ..>	_id_2	Barra deslizable para controlar la salida proporcional “Luz 2”. Este control se agrega por medio del comando “04”.
<h3>Persianas</h3>		Título de la segunda área “Persianas”. Este control se agrega por medio del comando “04”.
<input type=’range’ ..>	_id_3	Barra deslizable para controlar la salida proporcional “Persiana 1”. Este control se agrega por medio del comando “04”.
<input type=’range’ ..>	_id_4	Barra deslizable para controlar la salida proporcional “Persiana 2”. Este control se agrega por medio del comando “04”.

<code>&lt;h3&gt;Portón&lt;/h3&gt;</code>		Título de la tercer área “Portón”. Este control se agrega por medio del comando “04”.
<code>&lt;div id='_id5'.. &gt;&lt;/div&gt;</code>	<code>_id_5</code>	Visor que indica en que estado se encuentra el portón (“abierto” o “cerrado”). Este control se agrega por medio del comando “03” con un tiempo de validez de 1 segundo.
<code>&lt;button id='_id6'.. &lt;/button&gt;</code>	<code>_id_6</code>	Botón para activar la salida “Abrir” del portón. Este control se agrega por medio del comando “01”.
<code>&lt;button id='_id7'.. &lt;/button&gt;</code>	<code>_id_7</code>	Botón para activar la salida “Cerrar” del portón. Este control se agrega por medio del comando “01”.
<code>&lt;h3&gt;Accesos&lt;/h3&gt;</code>		Título de la cuarta área “Accesos”. Este control se agrega por medio del comando “04”.
<code>&lt;div id='_id8'.. &gt;&lt;/div&gt;</code>	<code>_id_8</code>	Visor que indica en que estado se encuentra el acceso 1 (“abierto” o “cerrado”). Este control se agrega por medio del comando “03” con un tiempo de validez de 1 segundo.
<code>&lt;div id='_id9'.. &gt;&lt;/div&gt;</code>	<code>_id_9</code>	Visor que indica en que estado se encuentra el acceso 2 (“abierto” o “cerrado”). Este control se agrega por medio del comando “03” con un tiempo de validez de 1 segundo.
<code>&lt;input type='text' id='_id_A' style='display:none' &gt;&lt;/input&gt;</code>	<code>_id_A</code>	Caja de texto enlazado a los contenidos de “_id_1”, “_id_2”, “_id_3” y “_id_4”. Este control se agrega por medio del comando “0A” con una periodicidad de 1 segundo.

Tabla 3

Luego debemos especificar los comandos que deberán ser enviados a la aplicación BluApp. Estos comandos son los siguientes:

[00Domótica]

```
[04<h3>Luces</h3><input type='range' min='-10' max='10' id='_id_1' align='left' style='width:60%'>Luz 1</input><br><input type='range' min='-10' max='10' id='_id_2' align='left' style='width:60%'>Luz 2</input><br><h3>Persianas</h3><input type='range' min='-10' max='10' id='_id_3' align='left' style='width:60%'>Persiana 1</input><br><input type='range' min='-10' max='10' id='_id_4' align='left' style='width:60%'>Persiana 2</input><br><h3>Portón</h3>]
```

```
[03;Estado:;_id_5;2000;<div id='_id_5' style='width:50%' ></div> <br>]
```

```
[01;_id_6;abrir;undefined;<button id='_id_6' style='width:20%' >Abrir</button>]
```

```
[01;_id_7;cerrar;undefined;<button id='_id_7' style='width:20%' >Cerrar</button>]
```

```
[04<br><h3>Accesos</h3>]
```

```
[03;Acceso 1: ;_id_8;2000;<div id='_id_8'></div><br>]
```

```
[03;Acceso 2: ;_id_9;2000;<div id='_id_9'></div>]
```

```
[0A;_id_A;1000;_id_1;_id_2;_id_3;_id_4;<input type='text' id='_id_A' style='display:none' ></input>]
```

[05]

Se observa que al comienzo se envía un comando “00” para modificar el título de la aplicación BluApp a “Domótica” y se agrega código html por



medio del comando "04" para algunos títulos y entradas. Luego se van especificando botones, y visores por medio de comandos "01" y "03" respectivamente. Al final se agrega un comando "0A" para poder enlazar múltiples entradas (contenidos de "\_id\_1", "\_id\_2", "\_id\_3" y "\_id\_4") al identificador "\_id\_A" y por último se agrega un comando "05" para guardar permanentemente la aplicación.

Existen tres maneras de cargar la configuración domótica a la aplicación BluApp:

- 1) La primera vez que el sistema electrónico establezca comunicación con la aplicación BluApp, le envíe por el enlace serie los comandos indicados previamente
- 2) Editar localmente la aplicación BluApp entrando en configuración, expandiendo el sub menú "Editar localmente", editando en la caja de texto los comandos indicados previamente y luego pulsando el botón "RX"
- 3) Entrando al menú de ajuste y luego al sub menú de Ayuda y pulsando el botón "Agregar ejemplo 1"

En la figura 8 se puede observar como queda nuestra aplicación personalizada.



Figura 8

## Contacto

mcmappser@gmail.com