

**INDUSTRIALES**  
ETSII | UPM

Trabajo Fin de Grado:

**DESARROLLO DEL  
PROYECTO  
ESCENOGRAFICO DE UNA  
PRODUCCIÓN MUSICAL**

Trabajo realizado por:  
Miguel Bernal Álvarez

---

Director: Félix Moreno

Madrid, Septiembre 2016





*Dedicado a todas las personas  
que han caminando a mi lado  
durante estos cuatro años.  
Gracias por creer en mí.*



## AGRADECIMIENTOS

Antes de comenzar con el desarrollo de este proyecto, me gustaría dar las gracias a todas personas que están cerca de mí:

En primer lugar, muchas gracias a mis padres por hacer posible mi formación como ingeniero y como persona. También quería agradecer a mis hermanos Cristina y Alfonso haberme acompañado todos estos años. Un agradecimiento especial a Miguel Ángel, mi otro hermano mayor.

Me gustaría dar las gracias a mis compañeros de la Escuela por todas las alegrías y sufrimientos compartidos. Confiad en mí, todavía nos queda mucho por vivir todos juntos.

Gracias a mi tutor, Félix Moreno, por creer en este proyecto y darme la oportunidad de llevarlo a cabo.

Gracias también a Óscar Gómez, por haber estado siempre dispuesto a echarme un cable, en el aspecto figurado y literal.

Para concluir, me gustaría dar las gracias a todos mis amigos, me enseñáis a disfrutar de la vida como se merece.

Miguel Bernal Álvarez.

# Resumen

El presente TFG tiene como objetivo diseñar, construir y poner en marcha un sistema de iluminación que servirá como escenografía en una producción musical.

Algunas de los números musicales del espectáculo se desarrollan en una gran iglesia y, por tanto, su escenografía se compone de cuatro grandes vidrieras de 4,5 metros de alto y un imponente rosetón de casi 3 metros de diámetro. Lo particular de estas estructuras es que cuentan con un sistema de iluminación que permite colorear de manera independiente las diferentes secciones que las forman. Además, las combinaciones de colores pretenden crear sorprendentes efectos visuales, lo cual se consigue a través de la sincronización de las luces con la música del espectáculo.

Con vistas a las necesidades de esta instalación, el proyecto trata de proponer una solución factible que posteriormente pueda ser implementada y utilizada por un grupo de teatro amateur durante sus representaciones en otoño de 2016.

Para diseñar este sistema, en primer lugar se realiza un estudio que dé a conocer herramientas y tecnologías que pueden satisfacer las necesidades de la instalación. Además, se debe analizar en profundidad el resultado esperado para que el producto obtenido sea lo más fiel posible.

Tras estos estudios iniciales, se elige la Tecnología LED RGB como sistema de iluminación y el protocolo WiFi como canal de las comunicaciones para lograr la sincronización total. Los microprocesadores escogidos, NodeMCU ESP8266, pueden ser programados a través del IDE Arduino. Además, incluyen un módulo WiFi capaz de enviar mensajes de texto a través del protocolo UDP o actuar como servidores de un sitio Web, entre otras posibilidades.

Una vez que se han seleccionado las tecnologías que van a fundamentar el funcionamiento de la instalación, es necesario aprender a implementarlas para ser capaces de elaborar un diseño eficaz y robusto. Por ello, en la presente Memoria se ha procurado alternar secciones que explican los principios básicos de estas herramientas y su aplicación concreta en el proyecto.

Durante los capítulos centrales del proyecto se definen en detalle los elementos que componen y permiten el funcionamiento correcto del sistema. Entre estos, el establecimiento de una red inalámbrica y el desarrollo del software que utilizan los microprocesadores. Para entablar comunicaciones efectivas entre los microprocesadores, se desarrolla una red que permite a un módulo maestro dirigir y coordinar al resto de módulos esclavos que, a su vez, gobiernan la iluminación LED RGB.

Con el objetivo de dotar al usuario con un mecanismo de control sobre la instalación, se decide desarrollar una interfaz gráfica que permita activar las diferentes rutinas de iluminación programadas. Se opta entonces por utilizar el

lenguaje HTML para sacar partido del servidor Web que los propios módulos WiFi pueden soportar.

Para concluir, se realiza un análisis sobre los resultados obtenidos y se establecen algunas posibles líneas de desarrollo futuro a fin de mejorar la solución propuesta. En términos generales, dichos resultados se pueden considerar satisfactorios ya que el sistema cumple todos los requisitos predefinidos al inicio del proyecto.

En último lugar, se presenta la planificación temporal del proyecto a través de un diagrama EDP y un diagrama de Gantt. Además, se muestra la ejecución de los presupuestos asociados a la construcción del sistema de iluminación así como la instalación de las estructuras.

### **Palabras clave**

Tecnología LED RGB, protocolo UDP, comunicación inalámbrica, iluminación artificial, Ingeniería del espectáculo.

### **Códigos Unesco**

Radiación Infrarroja, Visible y Ultravioleta (2202.06).

Colorimetría (2209.03).

Iluminación Eléctrica (3306.03).

Comunicaciones por WiFi y cable de red (3325.99).

Dispositivos de Transmisión de Datos (3304.12).

Sistemas de Reconocimiento de Caracteres (3304.06).

Teatro(6203.10).

# Índice general

## AGRADECIMIENTOS

RESUMEN.....	II
ÍNDICE.....	IV
ÍNDICE DE FIGURAS Y TABLAS.....	VI
LISTADO DE SIGLAS Y ACRÓNIMOS.....	VIII

## 1. INTRODUCCIÓN

1.1 MOTIVACIÓN.....	4
1.2 ESTADO DE LA CUESTIÓN.....	5

## 2. OBJETIVOS Y HERRAMIENTAS

2.1 OBJETIVOS DEL PROYECTO.....	8
2.2 HERRAMIENTAS.....	9
2.2.1 Tecnología LED RGB.....	9
2.2.2 IDE Arduino.....	16
2.2.3 Protocolo de Comunicación Inalámbrica.....	18
2.2.4 NodeMCU ESP8266.....	20

## 3. HARDWARE

3.1 COMPONENTES ELECTRÓNICOS.....	24
3.2 CIRCUITO DE ALIMENTACIÓN Y CONTROL.....	29

## 4. MONTAJE DE LA INSTALACIÓN

4.1 ESTRUCTURAS Y REALIZACIÓN FÍSICA.....	34
4.2 ASPECTOS ESTÉTICOS.....	38
4.3 INSTALACIÓN EN EL TEATRO.....	39

## 5. RED DE COMUNICACIÓN INALÁMBRICA

5.1 CARACTERÍSTICAS DE LA RED.....	42
5.2 ESTABLECIMIENTO DE LA INTRANET.....	47

## 6. DESARROLLO SOFTWARE

6.1 INTRODUCCIÓN AL DESARROLLO DEL SOFTWARE.....	51
6.2 PUESTA A PUNTO DE LAS HERRAMIENTAS.....	52
6.3 DISEÑO DEL SOFTWARE. FLUJOGRAMA.....	55
6.4 IMPLEMENTACIÓN DEL SOFTWARE.....	58
6.4.1 Programa Maestro.....	59
6.4.2 Programa Esclavo.....	63

<b>7. INTERFAZ GRÁFICA DE CONTROL</b>	
7.1 DISEÑO Y PROGRAMACIÓN DE LA INTERFAZ.....	66
7.2 FUNCIONAMIENTO DE LA INTERFAZ.....	68
<b>8. RESULTADOS, CONCLUSIONES Y LÍNEAS FUTURAS</b>	
8.1 RESULTADOS Y CONCLUSIONES.....	71
8.2 LÍNEAS DE DESARROLLO FUTURAS.....	73
<b>9. GESTIÓN DEL PROYECTO</b>	
9.1 DIAGRAMAS DE PLANIFICACIÓN TEMPORAL	
9.1.1 Diagrama EDP.....	75
9.1.2 Diagrama de Gantt.....	75
9.2 PRESUPUESTO.....	78
<b>BIBLIOGRAFÍA.....</b>	<b>80</b>
<b>APÉNDICES</b>	
A.1 CÓDIGO.....	83
A.2 PROCESO DE CONSTRUCCIÓN.....	84
A.3 ESQUEMÁTICO DE NODEMCU ESP8266.....	90

# Índice de Figuras y Tablas

<b>Figura 1:</b> Modelo 3D que representa la propuesta del proyecto .....	8
<b>Figura 2:</b> Bombilla incandescente.....	9
<b>Figura 3:</b> Bombilla fluorescente .....	10
<b>Figura 4:</b> Diodo LED .....	10
<b>Figura 5:</b> Espectro electromagnético y relación entre frecuencia y longitud de onda.....	11
<b>Figura 6:</b> LED RGB.....	12
<b>Figura 7:</b> Representación del modelo aditivo de colores RGB.....	12
<b>Figura 8:</b> Técnica PWM y Ciclo de trabajo.....	13
<b>Figura 9:</b> Implementación de la técnica PWM para obtener el color naranja.....	14
<b>Figura 10:</b> Muestra de tira LED RGB.....	15
<b>Figura 11:</b> Ventana de trabajo del IDE Arduino.....	16
<b>Figura 12:</b> Distribución de PINES de la placa NodeMCU DevKiT v2.0. ....	21
<b>Figura 13:</b> NodeMCU DevKiT v2.0 que incluye un microprocesador y módulo WiFi ESP8266.....	22
<b>Figura 14:</b> Consumo eléctrico asociado a los diferentes componentes del proyecto.....	25
<b>Figura 15:</b> Transistor TIP120 y el esquemático de su circuito: Colector, Emisor y Base.....	26
<b>Figura 16:</b> Circuito esquemático de los drivers utilizados en el proyecto.....	27
<b>Figura 17:</b> Drivers empleados para excitar las tiras LED del proyecto.....	28
<b>Figura 18:</b> Esquema que representa los cables de alimentación de los elementos vidriera y rosetón.....	29
<b>Figura 19:</b> Esquema que representa los cables de alimentación de los elementos vidriera y rosetón.....	30
<b>Figura 20:</b> Unidades necesarias de todos los circuitos de alimentación del proyecto.....	31
Escuela Técnica Superior de Ingenieros Industriales (UPM)	VI

<b>Figura 21:</b> Esquema definitivo del circuito de la instalación .....	32
<b>Figura 22:</b> Extremo superior de una vidriera.....	35
<b>Figura 23:</b> Ilustración esquemática de una vidriera con las tiras LED y por detrás, con las tapas de madera.....	36
<b>Figura 24:</b> Ilustración esquemática de los circuitos y su disposición en la parte trasera de una vidriera.....	37
<b>Figura 25:</b> Imagen para comparar la estructura del rosetón sin el papel difusor y con él.....	38
<b>Figura 26:</b> Fotomontaje del escenario que representa la instalación del sistema en el teatro.....	40
<b>Figura 27:</b> Topologías de Red, de izquierda a derecha: anillo, estrella y bus....	43
<b>Figura 28:</b> Explicación gráfica sobre la formación de una dirección IP de 32 bits.....	44
<b>Figura 29:</b> Clases IP.....	44
<b>Figura 30:</b> Formato de la cabecera del protocolo UDP.....	46
<b>Figura 31:</b> Modelo de la red 192.168.1.0.....	46
<b>Figura 32:</b> Diseño conceptual del programa del Módulo Maestro.....	56
<b>Figura 33:</b> Diseño conceptual del programa de un Módulo Esclavo.....	57
<b>Figura 34:</b> Lectura de la solicitud del cliente. Selección de la rutina.....	60
<b>Figura 35:</b> Notación de índices para la programación de colores en las estructuras.....	61
<b>Figura 36:</b> Dirección Web de la Interfaz Gráfica de Control.....	68
<b>Figura 37:</b> Interfaz Gráfica de Control.....	69
<b>Figura 38:</b> Diagrama EDP del proyecto.....	75
<b>Figura 39:</b> Tabla de actividades y su temporización.....	76
<b>Figura 40:</b> Diagrama de Gantt del proyecto .....	76

# Índice de Siglas y Acrónimos

**TFG**, Trabajo Fin de Grado

**LED**, Light-Emitting Diode

**IRC**, Índice de Reproducción Cromática

**RGB**, Red Green Blue. Rojo Verde Azul

**IDE**, Integrated Development Environment

**3D**, 3 Dimensiones

**Láser**, Light Amplification by Stimulated Emission of Radiation

**LCD**, Liquid Crystal Display

**PWM**, Pulse-Width Modulation

**ADC**, Analog-to-Digital Converter

**SMD**, Surface Mount Device

**WPAN**, Wireless Personal Area Network

**WLAN**, Wireless Local Area Network

**IEEE**, Institute of Electrical and Electronics Engineers

**UDP**, User Datagram Protocol

**IP**, Internet Protocol

**GPIO**, General Purpose Input/Output

**I2C**, Inter Integrated Circuit Communications

**SPI**, Serial-Peripheral interface

**GND**, Ground

**PC**, Personal Computer

**MDF**, Medium Density Fibreboard

**SSID**, Service Set Identifier

**TCP**, Transmission Control Protocol

Proyecto escenográfico de una producción musical

**WLAN**, Local Area Network

**DHCP**, Dynamic Host Configuration Protocol

**WPA-2**, Wi-Fi Protected Access 2

**AES**, Advanced Encryption Standard

**MAC**, Media Access Control

**USB**, Universal Serial Bus

**RSSI**, Received Signal Strength Indicator

**CPU**, Central Processing Unit

**URL**, Uniform Resource Locator

**HTML**, HyperText Markup Language

**CSS**, Cascading Style Sheets

**EDP**, Estructura de Descomposición del Proyecto

**IVA**, Impuesto sobre el Valor Añadido

Capítulo 1

# INTRODUCCIÓN

Motivación y Estado de la Cuestión

# 1. Introducción

Este Trabajo de Fin de Grado se incluye dentro del campo de la Ingeniería para el ocio. Se basa en la aplicación de principios técnicos fundamentales de la Electrónica, el Diseño del Producto y las Comunicaciones, entre otras áreas. En todos los campos de la Ingeniería, todos estos conocimientos científicos y técnicos se ponen al servicio del ser humano para hacer más sencilla su vida diaria. Esta es la verdadera razón de la Ingeniería.

El presente documento explica en profundidad el diseño y posterior construcción de un sistema ideado para satisfacer las necesidades de un espectáculo musical. Se trata de un producto de iluminación sincronizado con los diferentes números musicales que conforman dicho espectáculo.

Este producto, que exige un exhaustivo estudio de las técnicas en las que se fundamenta, será finalmente instalado en un teatro real y utilizado en más de una decena de representaciones en un grupo de teatro amateur de Guadalajara (España). El espectáculo es una adaptación de *Sister Act*, una comedia musical representada en una gran cantidad de países y basada en la homónima película de los noventa protagonizada por Whoopi Goldberg.

Si hay un rasgo que caracteriza este espectáculo, además de una inmejorable partitura compuesta por el célebre Alan Menken (*Aladdín*, *El Jorobado de Notre Dame*, *La Bella y la Bestia*, etc.), es su compleja y cuidada escenografía. En concreto, es remarcable el trabajo técnico que hay detrás de la iluminación de las escenas que se desarrollan en la iglesia del convento.

Este proyecto se basa en la iluminación de esas escenas, en la construcción de su escenografía y su puesta en funcionamiento para ser utilizada en las representaciones que tendrán lugar a partir de octubre de 2016.

La estructura de la presente Memoria se articula en los siguientes puntos, de tal modo que todos los ámbitos que el proyecto abarca quedan reflejados adecuadamente:

- i. Presentación de la problemática que se pretende solucionar incluyendo los objetivos perseguidos y las herramientas utilizadas.
- ii. Cuerpo central del proyecto que define en profundidad todos los conocimientos técnicos aplicados y las soluciones adoptadas.
- iii. Análisis de los resultados obtenidos tras el desarrollo del proyecto en comparación con las expectativas y el diseño realizado.
- iv. Gestión del proyecto: planificación temporal y económica.

## 1.1 Motivación

Un Trabajo de Fin de Grado tiene como último objetivo que los alumnos apliquen los conocimientos adquiridos durante su formación en las enseñanzas de Grado. Entre estos conocimientos se pueden encontrar nociones técnicas y científicas, pero también competencias transversales que determinan su crecimiento a nivel personal y profesional.

Tratándose de un Grado en Ingeniería, existe una infinidad de propuestas para elaborar estos proyectos. Sin embargo, se podría extraer un gran punto de convergencia entre todos los TFG desarrollados por los alumnos de cualquier formación técnica superior: su aplicabilidad.

En esto consiste la Ingeniería, de aplicar los complejos resultados físicos y matemáticos que estudian tan arduamente los científicos. Se trata de acercarlos a la gente para resolver los problemas que afectan a su vida cotidiana. Según Isaac Asimov: *"La ciencia puede divertirnos y fascinarnos, pero es la ingeniería la que cambia el mundo"*.

Por eso es tan amplia la oferta de TFG en las escuelas de Ingeniería, porque, lamentablemente, existe una gran variedad de problemas que afectan al mundo y que están esperando a que alguien los solucione. Entre estas situaciones se encuentran, por ejemplo: problemas de abastecimiento alimenticio, de bienes de consumo que hacen la vida diaria de la población mundial más sencilla o dificultades relacionadas con la salud de todos los seres humanos.

El presente proyecto se enmarca dentro del área de la Ingeniería para el ocio o el entretenimiento. [1]Si se observa con atención, la importancia de tener tiempo libre y ocuparlo en actividades que colaboran a la formación personal, tiene además un carácter preventivo de algunos de los males que amenazan a la sociedad: estrés, depresión, aislamiento, adicciones o enfermedades provocadas por el sedentarismo.

[2]Entonces se podría afirmar que el ocio es también salud, ya que el ejercicio de actividades ajenas a las obligaciones cotidianas mejora la calidad de vida de las personas. De acuerdo con recientes investigaciones sobre el entretenimiento, el ocio va más allá del tiempo libre y de las actividades que se realizan durante este tiempo. Más bien, se trata de un estado subjetivo o experiencia psicológica producida por la implicación en diferentes actividades. Este aspecto es crucial para reforzar la salud mental de aquellas personas que lo practican.

Pues bien, el fin último del producto es hacer disfrutar al público del espectáculo contribuyendo a su bienestar. Creando un ambiente festivo y alegre, la instalación permitirá a los espectadores dejar sus problemas y preocupaciones a un lado para centrarse en lo que sucede en el escenario.

Además de los beneficios directos sobre los consumidores de este tipo de ocio, otra de las razones de ser del proyecto es que proporciona un espacio de trabajo

creativo en el que aplicar los conocimientos adquiridos durante estos cuatro años.

Teniendo en cuenta la variedad de los conocimientos técnicos que se exigen para el diseño y realización de la instalación, este TFG toca una gran cantidad de materias relacionadas con las asignaturas del Grado. Entre ellas: Electrónica de Potencia, Microprocesadores, Dibujo Industrial asistido o Diseño de Webs Dinámicas.

Como conclusión personal de este apartado de motivación, además de todo lo aprendido durante el desarrollo del presente TFG, su elaboración ha abierto un nuevo campo donde llevar a cabo mi futura actividad profesional.

## 1.2 Estado de la Cuestión

A fin de llevar a cabo este proyecto, es necesario realizar un análisis previo sobre la situación actual de las tecnologías y su aplicación al mundo del ocio. El conocimiento del estado de las investigaciones permitirá tomar adecuadamente las decisiones que aparezcan durante la elaboración del TFG. Además, concede la oportunidad de visualizar los límites que hay en la actualidad y el modo de proceder de los ingenieros profesionales.

El presente apartado pretende esbozar sin mucho detalle la situación de las técnicas, ya que estos aspectos se trabajarán en profundidad durante el capítulo de herramientas. Sin embargo, sí se pueden encontrar tres bloques relacionados con el proyecto en los que resultará interesante centrar la investigación:

- i. Tecnologías de iluminación artificial en el presente.
- ii. Desarrollo actual de las comunicaciones.
- iii. Aplicaciones en el campo del ocio.

[3]En la actualidad existe una gran variedad de sistemas de iluminación artificial que se fundamentan en diversos principios para su funcionamiento. Entre todos, los más extendidos son las luminarias incandescentes y las lámparas de descarga.

Sin embargo, la iluminación artificial ha sufrido una revolución tras la llegada de un sistema de mayor rendimiento: los LED de luz blanca. Están muy bien posicionados para sustituir a las bombillas tradicionales. Algunas de sus ventajas son su alta fiabilidad, su larguísima vida útil y su IRC de, aproximadamente, el 90%. [4] El IRC (Índice de Reproducción Cromática) mide la capacidad de una fuente de luz blanca para mostrar los colores auténticos de los objetos cuando están siendo iluminados por dicha fuente. Sus valores se encuentran entre 0 y 100 y compara dichos colores con los visualizados cuando el objeto es iluminado con luz natural. Este alta tasa de IRC ha resultado

## 1. Introducción

decisiva para la generalización del uso de la tecnología LED en galerías de arte, museos o restaurantes.

Además, el impacto ambiental asociado a la fabricación y uso de la tecnología LED es considerablemente inferior en comparación con el provocado por la iluminación artificial tradicional: no contienen mercurio y emiten solamente luz perteneciente al espectro visible, frenando las emisiones infrarrojas y ultravioletas.

En segundo lugar se pretende estudiar la situación actual en el ámbito de las comunicaciones. Los avances tecnológicos de los últimos años han propiciado la creación de una sociedad de la información. La conexión a Internet ha establecido la posibilidad de compartir, opinar y responder a la ingente cantidad de información que llega desde el exterior. Este hecho ha revolucionado profundamente las relaciones humanas pero también ha cambiado la concepción de muchos sistemas de ingeniería.

El desarrollo de nuevos protocolos de comunicación permite su implantación en sistemas de producción, en infraestructuras civiles y en instituciones de toda índole. Por todas estas razones, en la actualidad se llevan a cabo una multitud de proyectos que se basan en la comunicación entre sus componentes. A fin de ilustrar esta información, se presenta el siguiente proyecto:

Una instalación de inteligencia artificial donde varios sensores recogen información sobre el cuerpo que atraviesa una calle. Cada uno elabora sus cálculos y concluye "qué o quién" ha cruzado la calzada. Gracias a la existencia de una red de comunicación, son capaces de enfrentar sus conclusiones y llegar a un acuerdo cuya fiabilidad rondará el 90%.

Además de estas estructuras de inteligencia artificial, el control de dispositivos a través de comunicaciones inalámbricas concede en la actualidad la posibilidad de realizar labores que entrañan peligro para el ser humano sin amenazar la integridad de ninguna persona. Por ejemplo, trabajos de rescate en zonas afectadas por catástrofes naturales o derrumbamientos.

Por tanto, a día de hoy no se puede hablar de una situación estática cuando se refiere a las comunicaciones. Se encuentran en pleno desarrollo y evolucionan continuamente para ofrecer cada vez más aplicaciones.

Para concluir con este capítulo de introducción, se pretende analizar brevemente la puesta en práctica de la Ingeniería en el mundo del ocio y, concretamente, del espectáculo.

En los últimos años se ha incrementado considerablemente la inversión tecnológica en productos relacionados con el entretenimiento. Desde la producción de megaconciertos hasta espectáculos musicales, es innegable que el mundo del ocio genera una actividad económica millonaria debido a sus necesidades técnicas. El control de sofisticados sistemas de iluminación, de sonido o incluso de maquinaria escénica se basa en los desarrollos tecnológicos más punteros y novedosos. En este último punto radica el interés y aplicabilidad del presente TFG.

Capítulo 2

# **OBJETIVOS Y HERRAMIENTAS**

Tecnología LED RGB, IDE Arduino,  
NodeMCU ESP8266 y Protocolos de  
comunicación inalámbrica

## 2.1 Objetivos

El presente apartado trata de explicar y definir con precisión cuáles son los objetivos del proyecto. Para abordar estos objetivos, se parte del producto que se quiere obtener. Las referencias del sistema de iluminación que se pretende construir vienen dadas por las soluciones adoptadas en las producciones profesionales del espectáculo. Sin embargo, los recursos disponibles son más escasos. Para el desarrollo de este proyecto, se cuenta con un presupuesto más reducido y un espacio escénico más pequeño, con las limitaciones que esto supone.

Dicho esto, el objetivo último de este TFG es diseñar y construir un conjunto escenográfico que incluya un sistema de iluminación dinámico sincronizado con la música del espectáculo. Para ello se deben satisfacer las siguientes especificaciones:

- i. Los elementos que se pretenden diseñar deben ambientar la escena como si sucediese en una iglesia. Estas estructuras son cuatro vidrieras y un rosetón, todos de grandes dimensiones.
- ii. Las cinco estructuras están divididas en paneles cuya iluminación debe ser independiente y programable.
- iii. La iluminación que se necesita para obtener los efectos deseados debe emitir luces de una gran variedad de tonalidades y colores.
- iv. La secuenciación de las combinaciones de colores no es aleatoria y, por tanto, necesita ser programada de acuerdo al desarrollo del espectáculo.
- v. Todas las luces deben cambiar simultáneamente y en sincronía con los ritmos de la música. Esto exige la necesidad del establecimiento de un sistema de comunicación.
- vi. Se necesita un espacio de comunicación entre el usuario y el sistema de tal modo que sea posible activar las rutinas de iluminación cuando sea pertinente.



Figura 1: Modelo 3D que representa la propuesta del proyecto

## 2.2 Herramientas

### 2.2.1 Tecnología LED RGB

La cuestión principal del proyecto es obtener un sistema de iluminación fiable y adecuado para resolver los requerimientos técnicos marcados en la propuesta teórica. Como se ha detallado en capítulos anteriores, los requisitos asociados a la iluminación son los siguientes:

- i. Amplia variedad de colores.
- ii. Bajo consumo energético.
- iii. Disipación térmica reducida.
- iv. Intensidad lumínica homogénea en superficie.

Atendiendo a las exigencias del proyecto, es necesario evaluar las diferentes posibilidades en cuanto a las luminarias o puntos de luz que se van a emplear. Las alternativas son diversas. [5]

Por un lado se encuentran las bombillas tradicionales o focos incandescentes. Estas luminarias basan su funcionamiento en un fenómeno conocido como termo radiación. En la termo radiación, la luz que se obtiene va siempre acompañada de una cuantiosa radiación térmica que supone una fuente de pérdidas energéticas cuando solamente se trata de producir luz. En la naturaleza también se puede observar este modo de producción de luz a gran escala. El Sol y las estrellas emiten grandes cantidades de energía lumínica y térmica. Artificialmente, las lámparas incandescentes producen luz por medio del calentamiento eléctrico de un filamento enrollado en doble espiral al vacío, normalmente de tungsteno o wolframio. En cuanto al espesor del hilo, dicho filamento es dos veces más fino que el cabello humano. Las luminarias que emplean estas bombillas alcanzan altas temperaturas y emiten una luz blanca (2700°K). Esta cantidad de grados Kelvin se utiliza para indicar el tono de luz emitido por un cuerpo. Se dice que emite una luz del mismo tono que un cuerpo negro calentado a esa temperatura.[6] Para modificar el color de esta luz sería necesario someter el cristal de la bombilla a diferentes tratamientos: acabado mate, coloración o espejado entre otros. [7]



Figura 1: Bombilla incandescente

Explicadas las características de las bombillas incandescentes, es necesario concluir que no son el sistema de iluminación más adecuado. Sin profundizar en exceso, estas bombillas no cumplen las especificaciones expuestas anteriormente: no son eficientes energéticamente y, además, producen grandes pérdidas de calor por efecto Joule. Este calor generado podría dañar el difusor óptico que se empleará como superficie visible de nuestra instalación. Además,

## 2. Objetivos y Herramientas

es absolutamente imposible extraer luces de diferentes colores sin el empleo de filtros adicionales o cristales tintados que fijan el color de cada bombilla. Esto es, no permiten que la misma luminaria emita luces de distinto color.

Por otro lado, se contempla también el uso de lámparas fluorescentes que basan su funcionamiento en la luminiscencia. Se trata de fenómenos luminosos que se producen cuando los electrones de una materia son incitados a producir



Figura 3: Bombilla fluorescente

radiaciones electromagnéticas. Es decir, a un átomo se le suministra una cantidad de energía que promueve el salto del electrón a una posición más energética (absorbe energía). Tras un breve tiempo, el electrón vuelve de manera espontánea a su posición original cediendo esa energía en forma de radiación electromagnética, principalmente radiación visible. La tecnología Láser (*Light Amplification by Stimulated Emission of Radiation*), por ejemplo, utiliza el fenómeno de la luminiscencia.

Las lámparas fluorescentes pueden ser obtenidas prácticamente en todas las características de color y en un amplio rango de tamaños y potencias. Sin embargo, también se trataría de un sistema fijo que no permite a una misma luminaria emitir luces de diferentes colores ni intensidades. Debido a este hecho, es necesario descartar esta tecnología para la iluminación.

Finalmente, el sistema adoptado para el desarrollo del proyecto será la tecnología LED RGB. Los diodos emisores de Luz (*Light-Emitting Diode*), o LED, funcionan por efecto fotoeléctrico. [8] Tienen dos patillas de conexión, una larga y otra corta. Para que pase la corriente y emita luz, se debe conectar la patilla larga al polo positivo (cátodo) y la corta al negativo (ánodo). En caso contrario, la corriente no pasa y no emite luz. Se dice entonces que los diodos LED emiten luz cuando se encuentran polarizados directamente. Su nacimiento tuvo lugar en la década de los sesenta, cuando tan solo se trabajaban en dos colores, rojo y verde. Estos diodos emiten en una determinada longitud de onda dependiendo del tipo de material semiconductor que los compone. En 1993 se desarrollaron también los LED de color azul y posteriormente blanco. Sus aplicaciones principales son, por ejemplo, la señalización electrónica en transporte, la retroiluminación LCD, los automóviles y, recientemente, la iluminación exterior e interior de edificios.

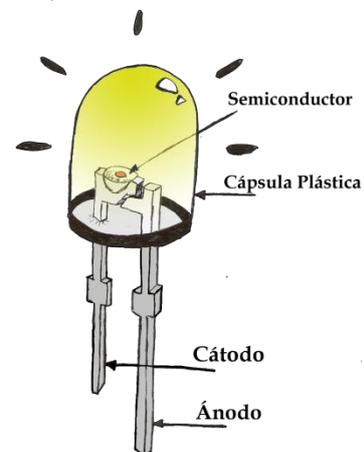


Figura 4: Diodo LED

Las ventajas de los diodos LED son entre otras su tamaño, su luminosidad y bajo consumo así como la posibilidad de atenuación de la intensidad lumínica.

Los diodos no son capaces de emitir luces de diferentes longitudes de onda, esta es siempre constante y por tanto, a priori no serían adecuados para conseguir la amplia gama de colores que necesitamos para nuestra aplicación. Entonces, con el empleo de iluminación LED se logra satisfacer tres de los cuatro requerimientos del sistema de iluminación. Los LED presentan un bajo consumo energético y, además, apenas disipan calor. Por otro lado, son ligeros y de pequeño tamaño, lo que permitirá que nuestra estructura-marco tenga un espesor reducido. Aunque en un principio el factor espacio no parece relevante, es esencial en una producción musical de estas características para un correcto y fluido desarrollo del espectáculo.

Sin embargo, el único requisito insatisfecho es la necesidad de una amplia gama de colores que permita crear diferentes efectos en el escenario. Este problema se resuelve utilizando el modelo de color RGB.[9] Este modelo se basa en la composición del color en términos de la intensidad de los colores primarios de la luz. Se conoce como un modelo aditivo de colores rojo, verde y azul. En inglés *Red, Green and Blue: RGB*. Estos colores son los tres colores primarios de la luz visible. La luz visible es blanca cuando se superponen todas las longitudes de onda del espectro visible ( $\lambda \in 390,750\text{nm}$ ), es decir, al sumar una luz roja ( $\lambda = 700\text{nm}$ ), una verde ( $\lambda = 546.1\text{nm}$ ) y otra azul ( $\lambda = 435.8\text{nm}$ ) se consigue luz blanca. No obstante, primero es necesario entender que este modelo nace del funcionamiento del sistema de visión humano.

Los ojos humanos tienen dos tipos de células sensibles a la luz o fotorreceptores: los bastones y los conos. Los bastones son sensibles a la luz, pero no al color. Sin embargo, los conos son menos sensibles a la luz, pero detectan el color. Estos últimos son los encargados de aportar esta información al sistema nervioso. Existen tres tipos de conos con respuestas diferentes, y que tienen máxima sensibilidad a los colores que forman la terna RGB. Es decir, cada uno de los tipos de conos que tenemos es más sensible a la radiación del rojo, verde o azul. Esta combinación permite percibir todos los colores del espectro, aproximadamente 10 millones de matices.

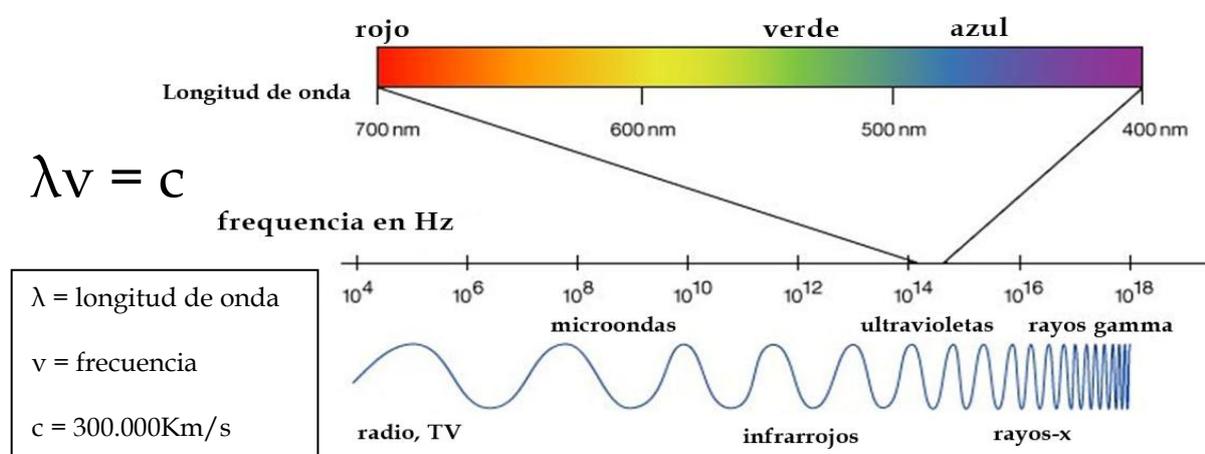


Figura 5: Espectro electromagnético y relación entre frecuencia y longitud de onda. Detalle ampliado del espectro de luz visible ( $\lambda$  entre 390nm y 700nm)

## 2. Objetivos y Herramientas

De este modo, la síntesis de estas tres señales construye el color con el que se percibe el objeto observado.

El modo de conseguir esta mezcla de colores RGB que el ojo humano es capaz de integrar como una luz de longitud de onda única es el uso de los llamados LED RGB.

Los LED RGB son diodos que tienen tres semiconductores. Cada uno presenta un color diferente. Esos tres colores son la terna asociada al RGB. Controlando esa mezcla de colores, se puede obtener una gama inmensa de luces de diferente longitud de onda. Es decir, dentro de cada LED RGB, se tienen tres pequeños focos de luz situados muy próximos unos de otros. Si todos emiten luz el ojo humano lo integra como un solo foco de luz monocromática, en este caso, blanco. Si se enciende un LED solamente, el resultado es una luz del color de ese LED (rojo, verde o azul). No obstante, al encender dos simultáneamente, se consiguen tres colores más: turquesa, amarillo y morado.

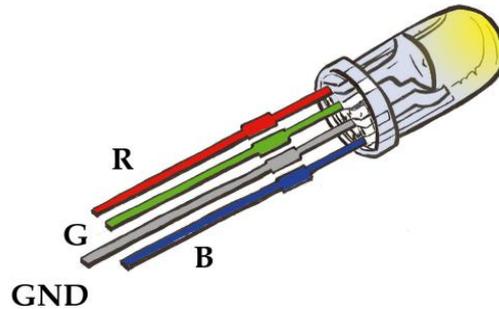
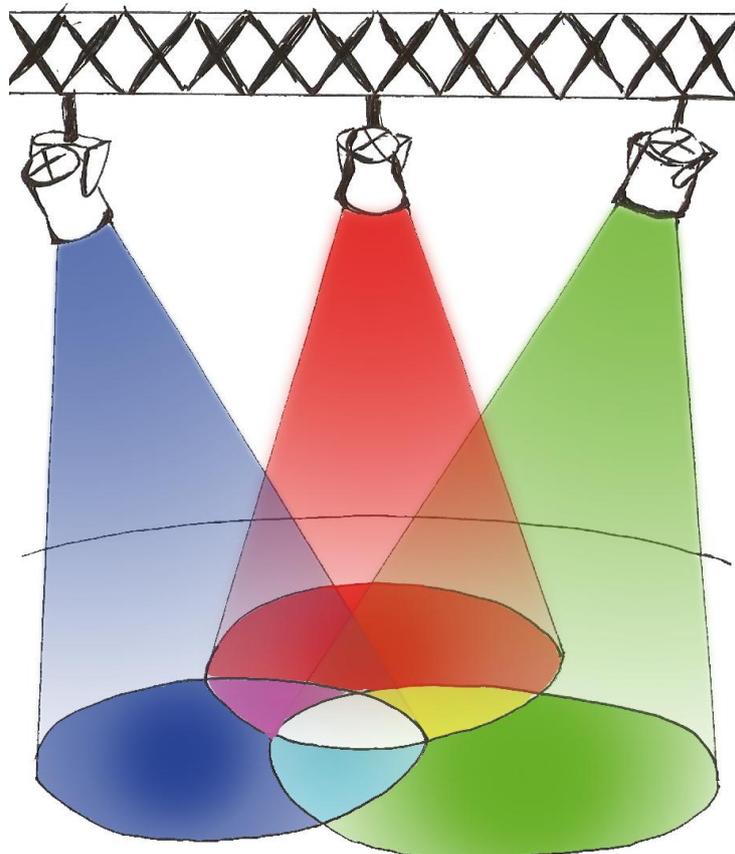


Figura 6: LED RGB

Figura 7: Representación del modelo aditivo de colores RGB



Ya se cuenta con una paleta de siete colores diferentes simplemente combinando diferentes configuraciones encendido-apagado en los tres diodos que forman el LED RGB. [10] Si ahora se introduce la técnica PWM (*Pulse-Width Modulation*) se puede controlar también la intensidad lumínica de cada diodo. Esto se traduce en una ingente cantidad de combinaciones diferentes de color, lo que implica una paleta de colores disponible prácticamente infinita.

El control PWM se realiza encendiendo y apagando el diodo a alta frecuencia de tal modo que el ojo humano integra el tiempo encendido y el apagado como si el LED se encontrase siempre emitiendo luz, pero a una intensidad diferente. Se conoce como ciclo de trabajo de una señal periódica al ancho relativo de su parte activa en relación con su período. Esta técnica es frecuentemente empleada para el control de motores de velocidad regulable o como parte de los convertidores analógico-digital ADC.

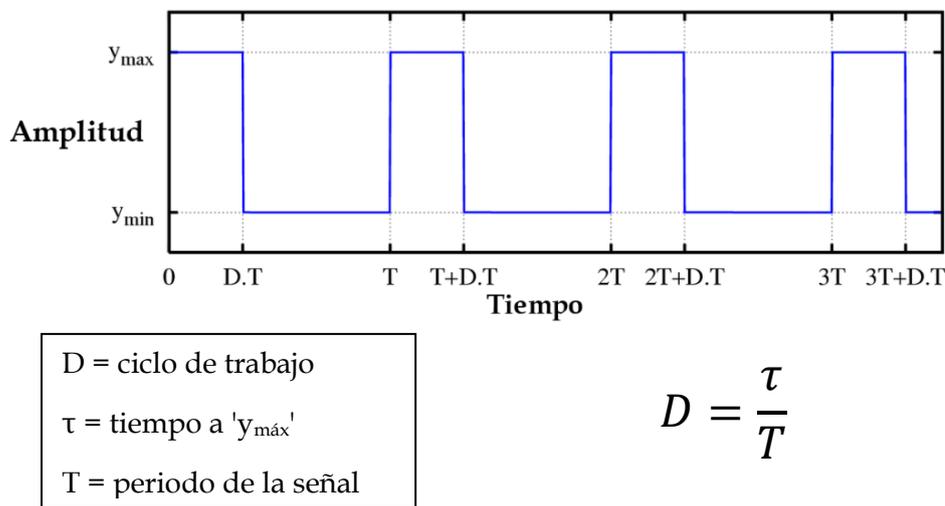


Figura 8: Técnica PWM y Ciclo de trabajo.

De este modo, regulando el ciclo de trabajo de cada diodo, se pueden conseguir intensidades lumínicas diferentes. Si se establece un ciclo de trabajo del 100%, se consigue una luz monocromática de máxima intensidad que se encuentra todo el tiempo emitiendo luz. Sin embargo, con un ciclo de trabajo del 50%, se alterna el tiempo que el diodo está emitiendo y sin emitir con una proporción 1:1. Con ello se logra una luz monocromática de intensidad media, encendida a media potencia.

Utilizando la técnica PWM se consigue una amplia gama de colores que permitirá en el proyecto ajustar muy finamente las tonalidades de luz que emitan las vidrieras y el rosetón. Además, los sistemas LED RGB son muy rápidos y no propiciarán el efecto fantasma que puede aparecer en otros sistemas. [11] Los LED se encienden instantáneamente al 100% de intensidad si es necesario y son más robustos ante las vibraciones, ya que se encuentran en

## 2. Objetivos y Herramientas

estado sólido. Gracias a todos estos factores, la iluminación LED RGB satisface con creces los requisitos imprescindibles de nuestro sistema de iluminación.

Volviendo a la relación entre la gama cromática obtenida y el control PWM, por ejemplo, obtener el color naranja mezclando rojo, azul y verde sí es posible. Sin embargo, no era realizable sin aplicar la técnica PWM ya que el ojo no es capaz de integrar ninguna combinación de los colores primarios con intensidad total como naranja. El color naranja se logra superponiendo una luz roja al 100% y una luz verde al 50%. Para esta mezcla, el diodo azul se encontraría apagado, al 0%.

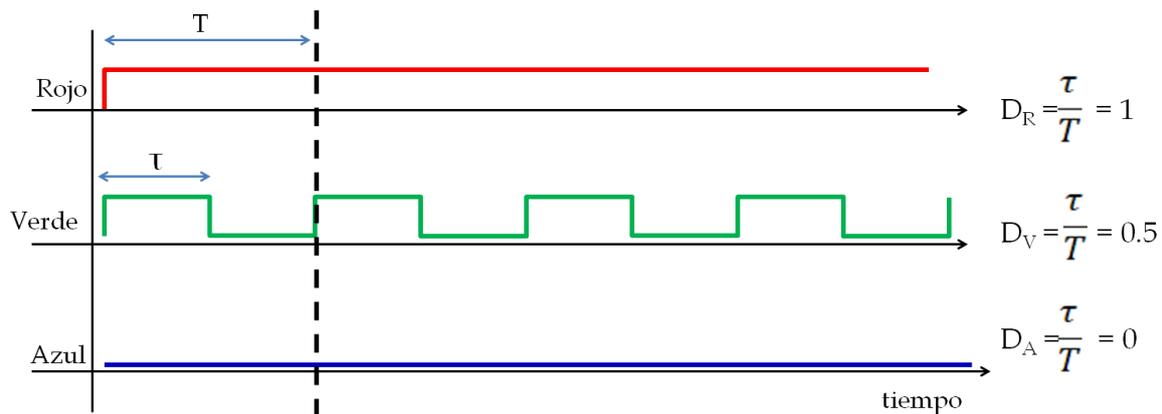


Figura 9: Implementación de la técnica PWM para obtener el color naranja

La técnica PWM se implementará en el circuito de control a través de la función *analogWrite* de la IDE Arduino. Este aspecto se desarrollará en profundidad en sucesivos capítulos.

Habiendo seleccionado ya los LED RGB como los dispositivos de iluminación que vamos a utilizar para nuestra instalación, hay que considerar su viabilidad técnica y el modo en el que se van a utilizar y posicionar dentro del sistema eléctrico.

Hoy en día se pueden encontrar muy fácilmente en el mercado, online o físico, una amplia variedad de productos relativos a la iluminación LED: desde componentes electrónicos de manera independiente hasta sistemas embebidos con su propio microprocesador. También existen equipos con su propia fuente de alimentación, un receptor de radio frecuencia y un control remoto a través del cual se puede seleccionar el color de emisión de los LED de una tira. Las tiras LED RGB ofrecen características bastante acordes a las solicitadas en este proyecto. Ofrecen una amplia variedad cromática, su control es inalámbrico, e incluso en algunos modelos su superficie trasera incorpora adhesivo para una fácil instalación en casi cualquier superficie. Además, es necesario tener en cuenta la flexibilidad y resistencia ante posibles golpes o dobleces.

Por supuesto, la integración de circuitos en la propia tira resulta su característica más destacable. Estas tiras contienen todas las conexiones y circuitos necesarios para el correcto funcionamiento de los LED incluyendo las resistencias pertinentes para evitar sobrecorrientes. Los LED se encuentran montados sobre las tiras como un Dispositivo de Montaje Superficial o SMD. Estas tiras se venden, normalmente, por metros. Otra de las ventajas de este producto es su longitud variable. Es decir, cada tres LED aparece señalado un área de corte. Si se separa la tira por esa sección, ningún componente sufre daño y las dos partes resultantes del corte se pueden utilizar sin fallo. Esto permite conseguir tiras LED de longitud variable para que se adapten perfectamente a la extensión que se desea ocupar.

Después de evaluar y considerar las diferentes posibilidades, estas tiras LED RGB serán la herramienta utilizada para la iluminación del proyecto.

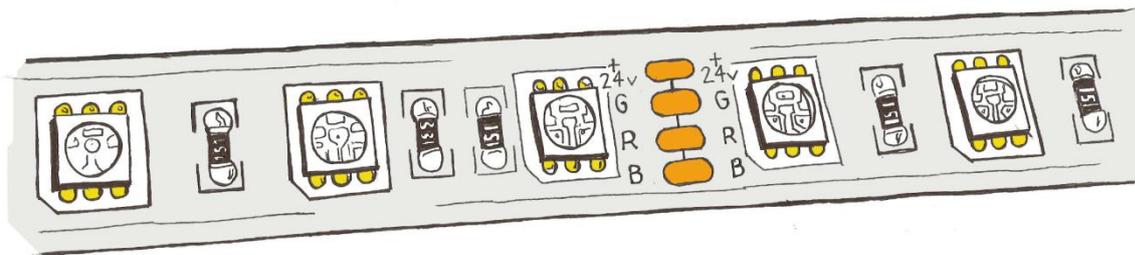


Figura 10: Muestra de tira LED RGB. La sección que permite su corte es el área en cobre coloreada en naranja

## 2.2.2 IDE Arduino

En primer lugar se van a explicar las principales razones que han determinado el empleo del IDE de Arduino como plataforma de desarrollo del Software de nuestra aplicación.

[12]Arduino es una plataforma de desarrollo de código abierto, basada en una placa con un sencillo microcontrolador y un entorno de trabajo para crear software o programas que serán cargados y posteriormente ejecutados en la placa. Se trata de un sistema versátil en el que, a través de unos puertos de Entrada/Salida, el microprocesador puede comunicarse con el exterior para recibir información de sensores e interruptores o también controlar multitud de tipos de luces, motores y otros actuadores físicos.

Las siglas IDE significan Entorno de Desarrollo Integrado (*Integrated Development Environment*) y se trata del software libre a través del cual se logra crear, abrir y modificar programas (*sketches*). Estos programas definen el trabajo que la placa debe realizar. Por tanto, este software incluye un editor de texto donde se puede escribir el código pertinente, un compilador que contiene un corrector de sintaxis y también la plataforma que permite cargar los programas a través de un puerto Serial. Además de cargar los programas, este puerto Serial habilita una comunicación entre el ordenador y el microprocesador realmente útil para la depuración del código (*debugging*).

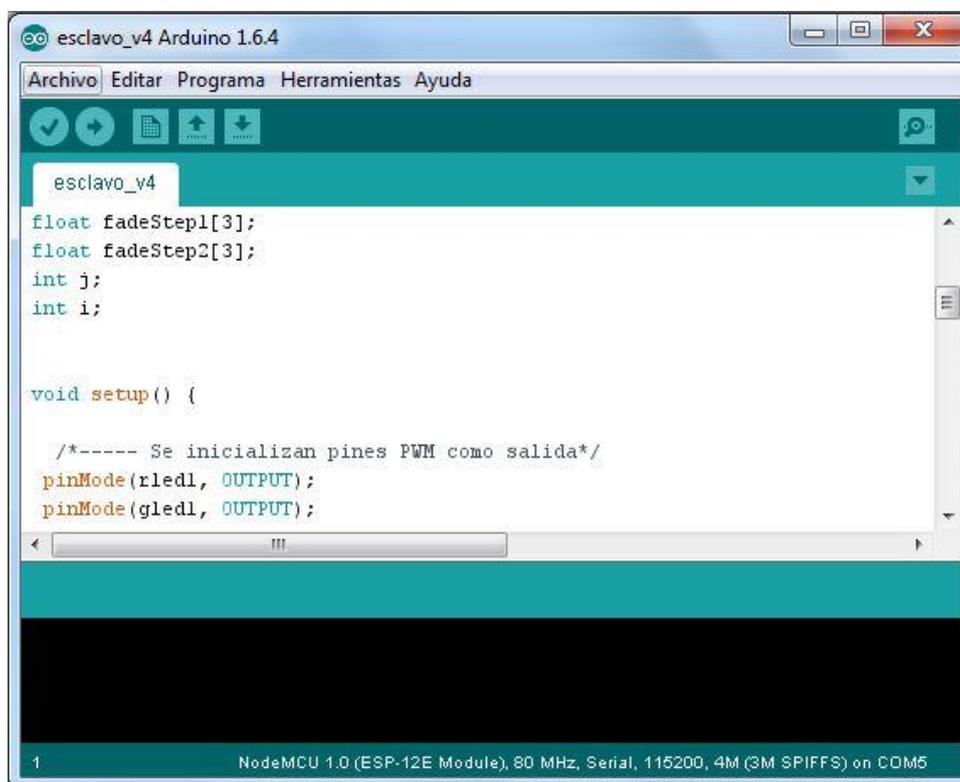


Figura 11: Ventana de trabajo del IDE Arduino. Inicialización de variables y setup() del código.

Las razones que han determinado el empleo de este tipo de microprocesadores son varias. En primer lugar, se trata de una plataforma libre con la que ya se ha trabajado en anteriores ocasiones. [13] Esto facilitará el desarrollo del código que regirá el comportamiento del circuito de control y las comunicaciones inalámbricas, que se explicarán más adelante. El lenguaje de programación es un lenguaje de alto nivel propio de esta plataforma. Sin embargo, resulta relativamente parecido al lenguaje de programación de alto nivel C. De hecho, la inclusión de algunas librerías desarrolladas por programadores "anónimos", enmascara algunos procesos de mayor complejidad. Esto permite desarrollar un código sencillo e intuitivo. Al tratarse de un software libre, hay librerías gratuitas a disposición de cualquier desarrollador de aplicaciones que las necesite.

Otra razón principal es el reducido coste de los microprocesadores que se van a utilizar para construir la instalación. Presentan un rendimiento considerablemente bueno frente a su precio, que no resulta especialmente elevado.

Para concluir con las razones que han motivado la elección de Arduino como plataforma de desarrollo, es necesario entender que además se perseguía una red de comunicación inalámbrica. Esta plataforma ofrece unas placas que integran un microprocesador y un sistema WiFi capaz de establecer dicha red. De este modo, el sistema podrá enviar y recibir información de un modo inalámbrico al mismo tiempo que procesa dicha información y actúa en consecuencia.

## 2.2.3 Protocolo de comunicación inalámbrica

De cara al requisito de sincronizar todos los comportamientos en cuanto a color e intensidad de las tiras LED RGB utilizadas en el proyecto, se consideró necesario establecer un sistema de comunicaciones eficaz que coordinase todos los microprocesadores empleados para su funcionamiento.

En un primer lugar, se evaluó la posibilidad de transmitir toda esa información a través de conexiones físicas entre los elementos del escenario que se iban a iluminar. Esta estrategia requería una gran cantidad de cables que, aunque se trata de un componente económico, aumenta considerablemente la complejidad de la instalación eléctrica y los problemas derivados de malos contactos y caídas de tensión asociadas a la longitud de los conductores. Además, si se considera la aplicación real de nuestro sistema, tantos metros de cable de un lado al otro del escenario podrían causar problemas durante el directo del espectáculo. Alguno de los actores o los elementos escenográficos podría dañar la instalación eléctrica o al contrario, siendo posible la lesión de algún miembro del elenco por la exposición a la corriente.

Por todas estas razones, se opta por un sistema de comunicación inalámbrica que permita enviar y recibir instrucciones para secuenciar las luces del montaje. De este modo se mejora en cuanto a versatilidad y flexibilidad del conjunto y se añade la posibilidad de iluminar elementos móviles en el escenario mientras que un sistema cableado fija las posiciones de la escenografía. Llegado este punto, se debe elegir entre distintas alternativas de comunicaciones inalámbricas.

Los protocolos para establecer redes de comunicación inalámbrica considerados para satisfacer nuestros requisitos son tres: ZigBee, Bluetooth o WiFi.

[14] ZigBee es el nombre de un conjunto de protocolos de alto nivel de comunicación inalámbrica de bajo consumo. Se basan en el estándar IEEE 802.15.4 para redes inalámbricas de área personal (WPAN). Su aplicación más recurrente en la actualidad es en domótica. Esto se debe a su bajo consumo, a su topología de red en malla que permite añadir nuevos nodos a la red de manera sencilla y a su fácil integración. [15] Bluetooth también pertenece a este tipo de redes WPAN pero, sin embargo, consume mayor cantidad de energía al lograr velocidades de descarga de hasta 3000kBs frente a los 250kBs de ZigBee. En cuanto a complejidad y cantidad de circuitería hardware, WiFi y Bluetooth son considerablemente más complejos que ZigBee, requiriendo este cerca de un 10% respecto a un nodo de Bluetooth o WiFi. La aplicación más común de los sistemas Bluetooth son los sistemas de telefonía y llamadas manos-libres debido a su velocidad. Pero el alcance de su red es menor y además admite menos nodos en su subred.

Por otra parte, [16] WiFi es el nombre que *Wi-Fi Alliance* dio al conjunto de estándares de comunicación 802.11 de IEEE. Una red de área local inalámbrica,

también conocida como WLAN (*Wireless Local Area Network*), es un sistema de comunicación inalámbrico que basa su funcionamiento en la transmisión de información a través de ondas de radio. La tecnología WiFi, que se encuentra considerablemente extendida en la actualidad para todo tipo de aplicaciones, va a constituir la base del sistema de comunicación inalámbrica de la instalación.

De cara a esta aplicación, se necesita transmitir información que indique a cada nodo de la red cómo debe comportarse. Esto es, un nodo maestro debe guiar al resto de nodos, de ahora en adelante esclavos, para ejecutar las distintas rutinas en sincronía. Esta información llegará a los esclavos en formato mensaje de texto. Por lo tanto, el módulo maestro preparará estos paquetes de datos y los envía a través de la red WiFi a la que todos los nodos se han conectado previamente. El protocolo UDP (*User Datagram Protocol*) se adapta perfectamente a esta especificación y, por tanto, será el canal de envío de información.

[17] UDP es un protocolo del nivel de transporte basado en el intercambio de datagramas que permite el envío de información sin previo establecimiento de la conexión. Esto se debe a que los paquetes de datos que envía contienen la información suficiente para el direccionamiento del paquete. Su sencillez en cuanto a la poca carga adicional que supone en la red, provoca, por otra parte, una disminución de su fiabilidad: UDP no emplea control del flujo de información. Sin embargo, gracias a no controlar la recepción del mensaje, se trata de un protocolo rápido, que permitirá a la aplicación enviar varios mensajes seguidos a alta velocidad sin que se aprecie la no simultaneidad.

El direccionamiento del paquete se realiza incluyendo la dirección IP del nodo al que se desea enviar dicho mensaje. Una dirección IP es un número que identifica de manera lógica, jerárquica e inequívoca a cada nodo de una red ya establecida. De este modo, la dirección IP es asignada por el creador de la red a cada nodo en el momento de conectarse. Entonces cualquier nodo puede enviar un paquete a cualquier otro nodo indicando su dirección IP.

## 2.2.4 ESP8266 ESP12-E NodeMCU

Una vez definida la plataforma de desarrollo software que se va a utilizar para llevar a cabo este proyecto y la tecnología de comunicaciones inalámbricas, es necesario decidir un modelo de microprocesador que reúna los requerimientos solicitados para la aplicación. Como se ha esbozado en anteriores apartados, nuestro sistema de control debe ser capaz de gestionar dos aspectos de la instalación:

- i. Gobierno y dirección de la iluminación en cuanto a términos de color e intensidad en las tiras LED RGB.
- ii. Establecimiento de una red de comunicación capaz de sincronizar adecuadamente los sucesivos efectos de un modo secuencial.

Se planteó el uso de diferentes modelos de microprocesadores como, por ejemplo, los de la casa *Raspberry Pi*. Sin embargo, estos microprocesadores resultan menos económicos, en general, que los que utilizan la ya seleccionada plataforma de desarrollo.

Atendiendo al primer requisito fundamental de nuestra aplicación y a lo explicado previamente acerca del control de las tiras LED RGB, se necesita un microprocesador con puertos de Entrada/Salida capaces de desarrollar la técnica PWM. Se recuerda que incluyendo esta prestación se consigue una cantidad considerablemente grande de colores distintos. Este número dependerá de la resolución del ciclo de trabajo que ofrezca el microprocesador elegido.

En cuanto a la necesidad de establecer una red de comunicación que permita la sincronización de la iluminación y la música, se opta por una red inalámbrica. En el anterior apartado se explican las razones que han motivado, finalmente, el empleo de la tecnología WiFi para establecer dicha red. Por tanto, nuestro microprocesador debe ser capaz de conectarse a esta red para el tránsito de información que secuencie todo el sistema. En los últimos años, se ha estandarizado el uso de un módulo WiFi bastante económico que permite comunicar diferentes microprocesadores, o un micro y un sensor o actuador para manejarlo. Este módulo se conoce como ESP8266.

Teniendo en cuenta estas dos solicitudes, lo más adecuado es emplear una placa que sea capaz de satisfacer los requisitos del proyecto. Para ello, se barajan diferentes posibilidades llegando a la solución adoptada.

[18]La placa de desarrollo NodeMCU combina características de *Access Point* (AP) y microcontrolador. Este microprocesador satisface entonces los dos requerimientos del sistema de control de la aplicación. Cuenta con 10 pines de Entrada/Salida de propósito general (10 GPIO), funcionalidad PWM, comunicación I2C y SPI y un conversor analógico-digital. Además, puede trabajar en una red WiFi tanto como Punto de Acceso como servidor Web.

Como característica adicional, también puede conectarse a internet para descargar o subir datos. La configuración de esta placa para poder utilizar la IDE Arduino es ligeramente laboriosa al principio, ya que hay que instalar drivers para la comunicación Serial a través de los puertos USB entre otras cosas. Sin embargo, una vez configurado adecuadamente, resulta muy sencillo utilizar y acceder a todas las prestaciones que ofrece dicho microprocesador.

A través de los GPIO (*General Purpose Input/Output*), vamos a ser capaz de gobernar las tiras LED RGB. Para ello necesitaremos configurar los pines digitales pertinentes como salida. Después se dará un valor a dicha señal. Obviamente, la pequeña señal que se puede extraer del microprocesador no es suficiente para alimentar las tiras y, por ello, se utilizarán unos drivers para alimentarlas a 12V, su tensión nominal. Cada tira que se quiera iluminar independientemente, ocupará tres pines digitales (rojo, verde, azul) más una toma de tierra GND.

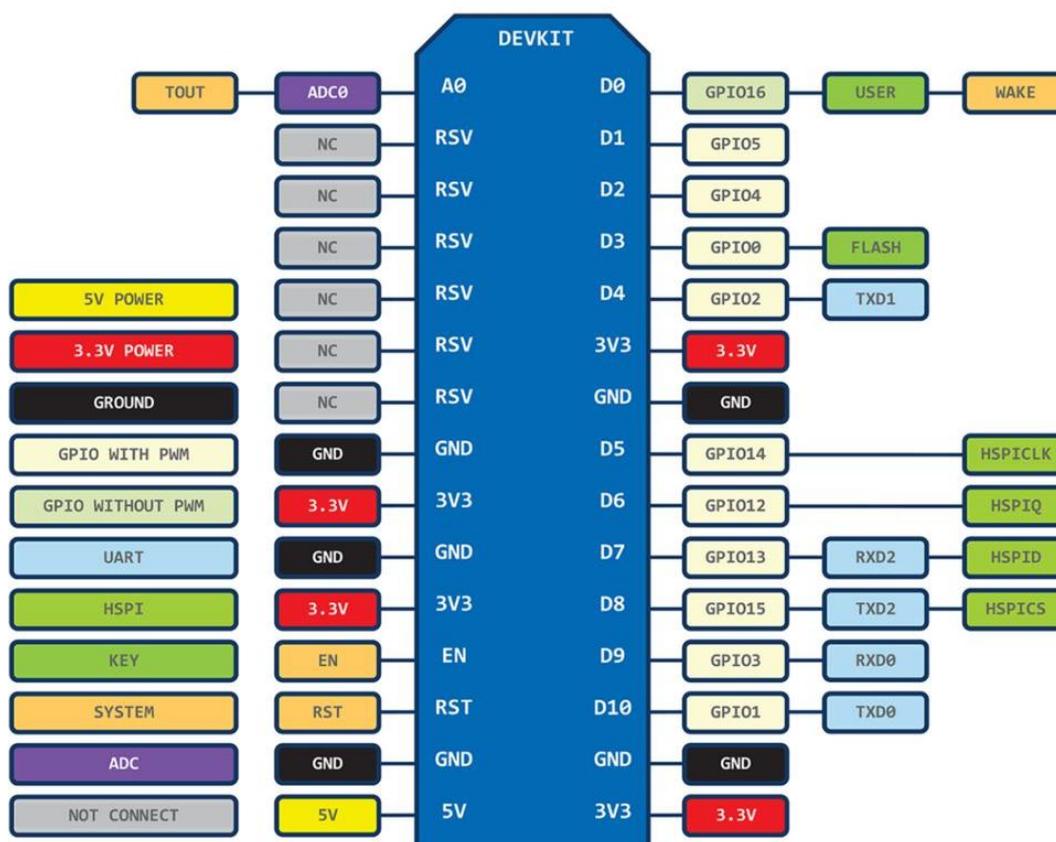


Figura 12: Distribución de PINES de la placa NodeMCU DevKit v2.0

En estos tres pines digitales se impondrá una señal digital dirigida a través de la técnica PWM, que ya se ha explicado en apartados anteriores. De este modo, el ciclo de trabajo, o *duty cycle*, de este microprocesador tiene una resolución de 8 bits. Eso implica que habrá  $2^8$  configuraciones distintas del PWM y por tanto 256 ciclos de trabajo distintos. Para aplicar PWM en el micro seleccionado, se utiliza la función *analogWrite(pin, value)*. Esta función escribe en los pines digitales HIGH o 1 durante el tiempo encendido de la señal periódica en

## 2. Objetivos y Herramientas

función de su argumento *value*. El valor indicado en ese argumento es un entero comprendido entre 0 y 255. 255 es el máximo valor que se puede escribir y por tanto indica que el pin se mantendrá encendido durante todo el periodo. Por ejemplo, si se desea que aparentemente la salida de ese pin sea 0.5, se debe introducir un 128 en la función. Esto equivaldría a un ciclo de trabajo del 50% y por tanto, el LED conectado al pin indicado se encontraría aparentemente encendido a media potencia.

Por otro lado no se puede olvidar el establecimiento de la comunicación WiFi. Para controlar este aspecto del sistema es necesario descargar una librería que ofrece distintas funciones. Esta librería permite un acceso sencillo a las diferentes funcionalidades del Módulo WiFi ESP8266. Entre ellas, se encuentran el protocolo de intercambio de mensajes UDP y la posibilidad de crear un Servidor Web.

Para este proyecto se van a aprovechar ambas prestaciones de estos módulos WiFi. Por un lado, se enviarán mensajes de texto a través del protocolo UDP para sincronizar todas las tiras LED RGB y además se desarrollará una interfaz gráfica de tipo página Web.

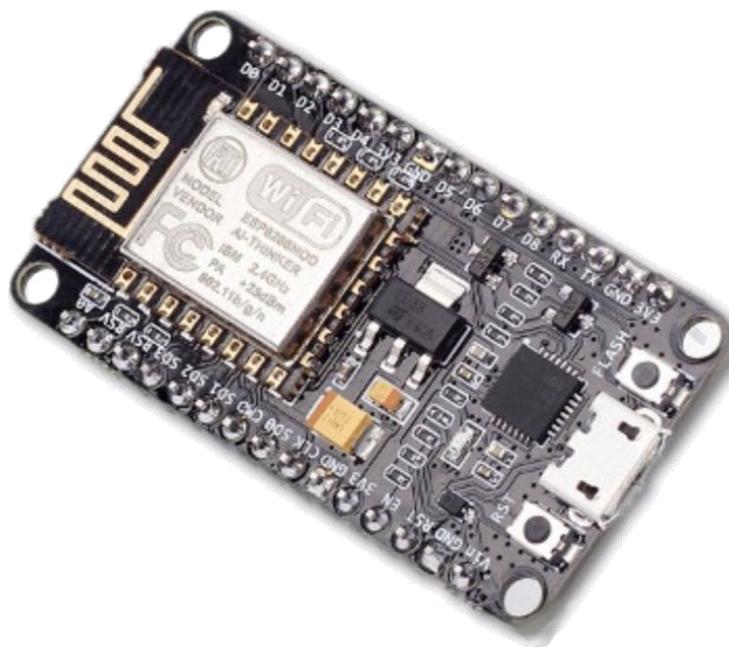


Figura 13: NodeMCU DevKit v2.0 que incluye un microprocesador y módulo WiFi ESP8266

## Capítulo 3

# **HARDWARE**

Componentes electrónicos, circuito de  
alimentación y control

## 3.1 Componentes electrónicos

En este apartado se pretende explicar los diferentes componentes eléctricos y electrónicos que conforman la instalación eléctrica del sistema.

En primer lugar, resulta obvia la necesidad de una fuente de alimentación que ceda potencia suficiente para iluminar todas las tiras LED RGB (*Véase capítulo 2.1*) y que, además, alimente a los controladores. Otro componente fundamental son los drivers. [19] Los microcontroladores no son capaces de manejar tensiones e intensidades más allá de los 5V (tensión de alimentación) y los 50mA, aproximadamente. Los actuadores del circuito requieren tensiones y corrientes que destruirían los circuitos digitales y, por tanto, es imprescindible un "traductor" entre la sección del circuito que requiere altas intensidades y las pequeñas señales que proceden del micro. Estos "traductores" se conocen como drivers y basan su funcionamiento en transistores y semiconductores que permiten, o no, el paso de energía en función de una señal de referencia o control. Además de la fuente, los microprocesadores y los drivers se emplearán otros componentes eléctricos como clemas (o borneras) para realizar las distintas conexiones, un interruptor diferencial y diferentes mangueras de cables para transportar la información y energía necesarias.

En electrónica, una fuente de alimentación es un dispositivo que convierte la tensión alterna de la red de suministro, en una o varias tensiones, prácticamente continuas, que alimentan o suministran energía a su salida. Para el sistema en cuestión, se necesita pasar de 240V y 50Hz a 12V continuos para las tiras LED y 5V también continuos para los microcontroladores. La fuente de alimentación que se utilizará en el sistema debe ser seleccionada entre distintas alternativas. Se barajan dos posibilidades: fuente de alimentación fija para tiras LED o fuente de alimentación de PC. Se pueden encontrar en el mercado fácilmente fuentes dedicadas exclusivamente a la alimentación de tiras LED que requieren 12 o 24V continuos.

Estas fuentes de alimentación se conocen como fuentes de alimentación fija y se ha descartado su uso en esta aplicación por dos factores. Por un lado, no se trata de una solución versátil ya que tan solo ofrecen un potencial (12 o 24V) cuando se necesitan además 5V para alimentar los microprocesadores. Esto implicaría obligatoriamente construir dos circuitos de alimentación independientes, con el consiguiente incremento del presupuesto y de la magnitud del cableado. La segunda razón que motiva desechar esta solución es el precio. Estas fuentes de alimentación para tiras LED se encuentran en el mercado a un precio superior que la segunda alternativa, las fuentes de alimentación para PC.

De esta manera, las fuentes de alimentación para PC se adaptan perfectamente a las necesidades del proyecto ofreciendo la posibilidad de extraer, con diferentes tensiones, la energía que requiere nuestro circuito.

Una vez seleccionado el tipo de fuente, es necesario dimensionar el consumo eléctrico del circuito general para conocer la energía que, al menos, debe ceder la fuente de alimentación elegida. Se va a realizar una estimación del consumo de cada elemento Para ello se tomarán las siguientes consideraciones:

- i. El consumo promedio según fabricante de una tira LED RGB con 60 LED/metro es de unos 72 W cada 5 metros.
- ii. La longitud necesaria de tiras LED para cada elemento depende de sus dimensiones (*Véase capítulo 4*).
- iii. Los módulos WiFi (NodeMCU ESP12-E) utilizados se alimentan a 5V y a través de ellos circula una pequeña corriente de aproximadamente 75mA en promedio. El número de micros empleados depende también del elemento: rosetón, vidrieras traseras o vidrieras delanteras.

	Metros de tira LED	Consumo por cada 5 metros de tira	Unidades de Módulos WiFi	Consumo unitario	Consumo General (W)
Vidriera delantera	35	6A a 12V 72 W	8	75 mA a 5V 375mW	507
Vidriera trasera	32		7		463.425
Rosetón	25		5		361.875

Consumo máximo (W)	507
--------------------	-----

Figura 14: Consumo eléctrico asociado a los diferentes componentes del proyecto.

Por tanto, se precisa de una fuente de alimentación capaz de ceder más de 507W. Es obligatorio sobredimensionar ligeramente esta cifra como factor de seguridad. Adquirir una fuente de 600W aproximadamente dará un margen relativamente amplio al circuito sin reducir el rendimiento de la misma. Como conclusión, será necesario adquirir una fuente de alimentación del orden de los 600W para cada elemento escenográfico que se vaya a iluminar. Gracias a la red de comunicación inalámbrica, el único cable que relacione el sistema con el exterior será el que lo conecte a la red eléctrica. Otro aspecto que hay que tener en cuenta es el reparto de la potencia cedida por la fuente. Existe un valor máximo para cada tensión nominal de salida. La fuente cede, por ejemplo, a 3.3V, 5V o  $\pm 12V$  y hay un amperaje máximo cedido por la misma para cada tensión. Esto implica que además de evaluar el consumo general eléctrico del sistema, hay que considerar el consumo en amperios para cada voltaje.

### 3. Hardware

Una vez resuelta la cuestión de la alimentación, se procede a explicar la implementación de los drivers como lugar de unión entre el circuito eléctrico que excita los LED y el control que se tiene sobre los mismos.

Un driver, o circuito de excitación de los LED de este proyecto, es un dispositivo capaz de regular la intensidad que circula a través del circuito. Los drivers utilizados en este proyecto contienen tres transistores tipo NPN para abrir o cerrar el flujo eléctrico que alimenta los LED de cada color (RGB). [20] Cuando un transistor funciona ante el paso de la corriente se dice que funciona al corte (no pasa) o a saturación (pasa sin restricciones). Este modo de funcionamiento es la base de la tecnología digital moderna. De hecho, un transistor en corte es un '0' y en saturación es un '1', FALSE/TRUE.

Los transistores cuentan con el Colector, el Emisor y la Base. Cuando se pone una señal eléctrica variable en la Base, el flujo de corriente entre el Emisor y el Colector sigue la señal de la misma pero amplificada. Para lograr esa funcionalidad, se emplean materiales semiconductores. Los semiconductores, típicamente Silicio y arseniuro de Galio, se convierten en conductores o aislantes mediante una señal eléctrica.

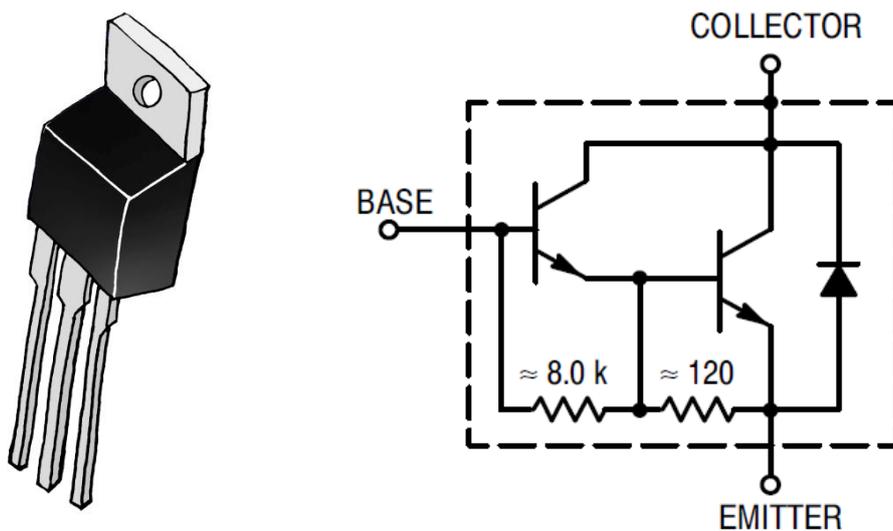


Figura 15: Transistor TIP120 y el esquemático de su circuito: Colector, Emisor y Base.

Gracias al empleo de estos transistores se logra controlar la intensidad que atraviesa las tiras LED, y por tanto, su emisión de luz. El colector se conecta a la fuente de alimentación, a 12V. Por otro lado, la señal eléctrica de la base proviene de un pin de salida del microprocesador NodeMCU utilizado en el proyecto. Atendiendo al valor de esta señal eléctrica, una pequeña señal, se permite el paso de una intensidad proporcional a la misma. Para finalizar, esta intensidad, que recorre la línea colector-emisor, es la que alimenta los LED.

Un transistor, por tanto, permite alimentar y controlar tan solo un color de los tres que contiene la tira que se pretende iluminar. Replicando este control para

los otros dos colores que conforman la tira se consigue gestionar totalmente la tonalidad de la luz que emiten los LED. Este es el principio sobre el que está fundamentado el funcionamiento de los driver de control de nuestro sistema.

El circuito de excitación de los LED cuenta con seis entradas: dos correspondientes a la alimentación y las otras cuatro asociadas al control. De las dos entradas de alimentación, una se encuentra a 12V (1) y la otra es la Masa (2) o referencia de tensión (GND). Esta segunda, como puede apreciarse en el esquemático del circuito, está cortocircuitada con la referencia de polaridad de las entradas de control (3). En cuanto a las otras tres entradas, se trata de las salidas con PWM del microprocesador que actúan como señal de referencia para la base de los transistores TIP120. Una para cada color: Rojo (4), Verde (5) y Azul (6).

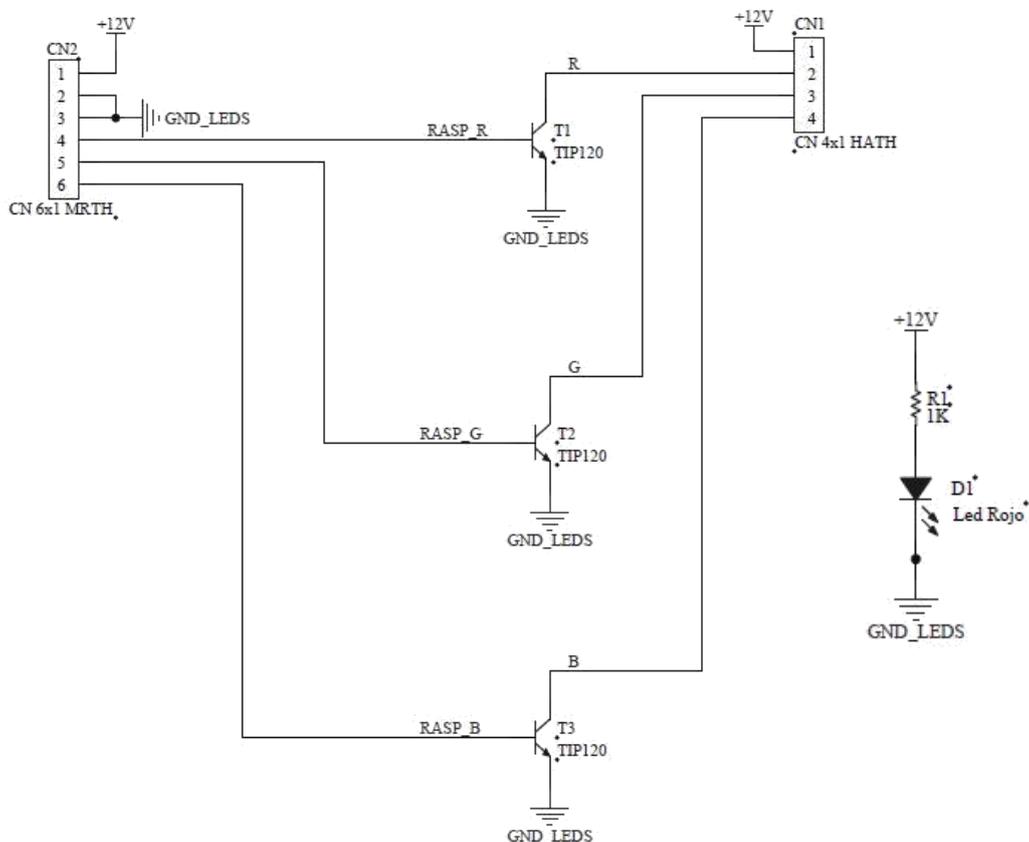


Figura 16: Circuito esquemático de los drivers utilizados en el proyecto

Por otro lado, las salidas del driver son cinco teniendo en cuenta el LED indicador, que aparece también en el esquemático. Las otras cuatro salidas conducen las intensidades que recorren los LED. La primera de ellas a 12V (1) y las otras tres corresponden a los colectores (2, 3 y 4) de los tres transistores TIP120. Estas cuatro salidas se conectan directamente a la tira LED a través de las cuatro secciones de cobre de la misma.

### 3. Hardware

La instalación de estos drivers es realmente sencilla gracias a su diseño compacto y a las clemas que porta para realizar las conexiones del cableado necesarias. Las dimensiones de la placa son 40x40mm e incluye tres taladros pasantes en sus esquinas para fijar su posición utilizando tornillos. Sin embargo, en este proyecto se utilizará otro sistema de fijación que no dañe el material soporte, ya que esto podría empeorar las condiciones de difusión de la luz.

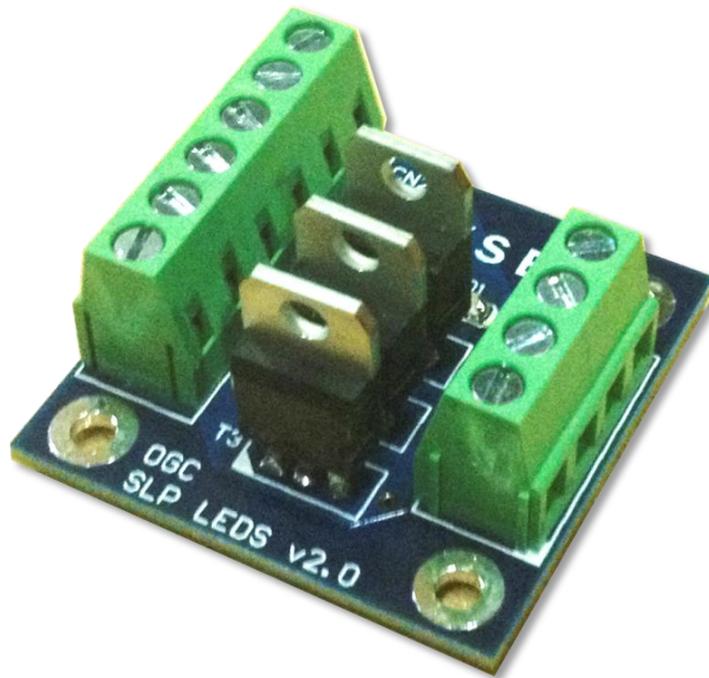


Figura 17: Drivers empleados para excitar las tiras LED del proyecto

## 3.2 Circuito de alimentación y control

Una vez descritos los componentes electrónicos que forman parte del circuito de alimentación de las tiras LED, se procede a detallar las relaciones entre las diversas partes de la electrónica de la instalación. Para ello es necesario establecer algunos requisitos imprescindibles del circuito eléctrico que se va a construir y así poder tomar una estrategia bien definida en su realización.

En primer lugar, ya que se van a utilizar comunicaciones inalámbricas, el número de cables de entrada al conjunto debe ser lo menor posible. De este modo se consigue simplificar la instalación de las vidrieras y el rosetón en el teatro. También se evitan posibles accidentes asociados a la presencia de cables por el escenario, donde se desarrollan en directo movimientos muy complejos tanto de elementos escenográficos como de personas. A fin de encontrar una solución a esta circunstancia, se decide acoplar en la parte trasera de cada vidriera toda la circuitería necesaria para su funcionamiento. El único elemento independiente a la misma es su fuente de alimentación.

Por tanto, los únicos cables que salen de cada elemento, vidriera o rosetón, son los de alimentación de los drivers a 12V y la alimentación de los módulos NodeMCU a 5V. Estos van directamente a su fuente de alimentación, conectada a la red eléctrica.

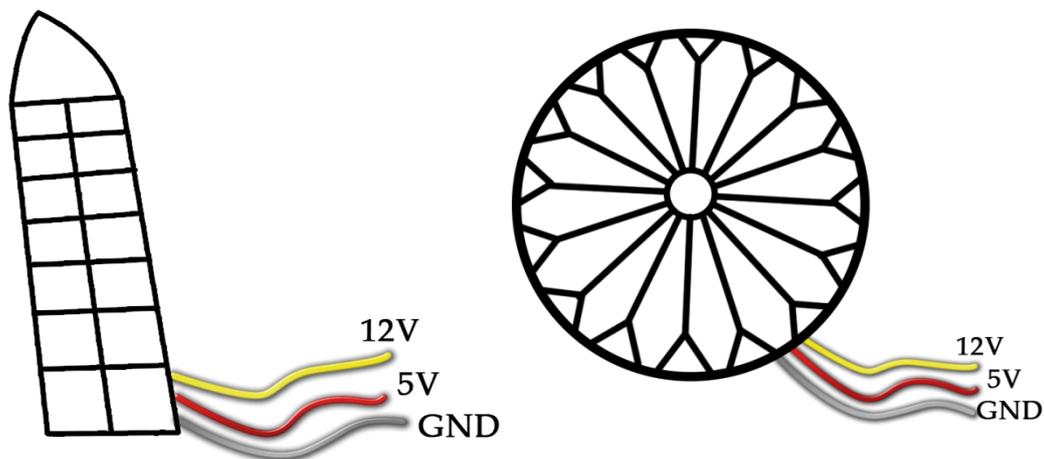


Figura 18: Esquema que representa los cables de alimentación de los elementos vidriera y rosetón

Como se ha especificado previamente, en este proyecto se tienen tres tipos diferentes de elemento escenográfico: vidriera delantera, vidriera trasera y rosetón. Sin embargo, la metodología para llevar a cabo su realización física es la misma y tan solo se diferencian en sus dimensiones y morfología. Esto implica que el número de componentes electrónicos necesarios para el rosetón es diferente que el de las vidrieras.

### 3. Hardware

De ahora en adelante, en este apartado se considera la construcción de una vidriera trasera ya que el esquema del proceso es idéntico en los tres casos.

Dicho esto, se definen dos alternativas para la construcción del circuito:

La primera estrategia que se estudió fue el empleo de un solo microprocesador por vidriera. La función de este módulo sería recibir las órdenes de cambios de colores por WiFi. Como se pretenden iluminar de manera independiente 15 tiras LED, y el módulo NodeMCU ESP12-E solo cuenta con 8 pines de salida configurables para PWM, sería necesario demultiplexar la señal de salida. Esto implica el uso de un decodificador que haga barridos con la salida del micro por todas las tiras que se quieren iluminar. [21]Un demultiplexor es un circuito combinacional integrado que permite activar tan solo una de las salidas del circuito en función de una señal de control gobernada por el mismo microprocesador.

Como la señal de salida es triple (RGB), la placa debería contener tres demultiplexores, uno para cada color. Estos demultiplexores deben ser de 1 a 16 para poder controlar los 15 drivers de las 15 tiras. Teniendo en cuenta las dimensiones del demultiplexor, se requiere una señal de control de 4 bits que permita seleccionar la salida que corresponda. Además, los 3 demultiplexores de la placa deben funcionar en sincronía para evitar incongruencias en los colores emitidos por cada tira LED.

El esquema del circuito completo sería el siguiente:

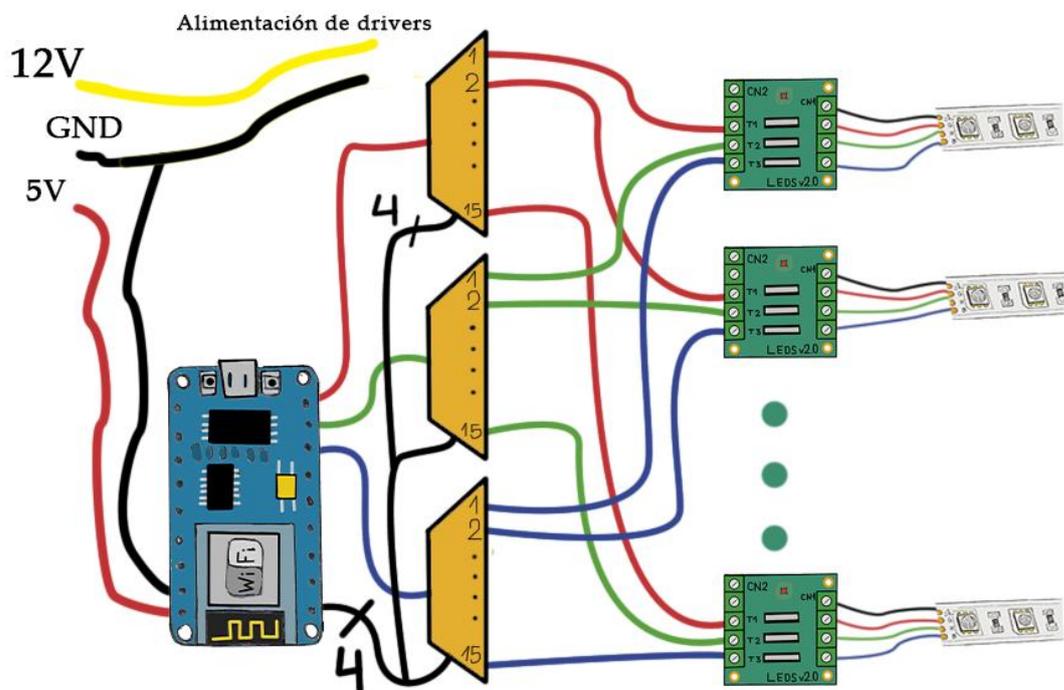


Figura 19: Esquema que representa los cables de alimentación de los elementos vidriera y rosetón

Utilizando el circuito propuesto, el microprocesador debe ir realmente rápido refrescando todos los drivers para que no se vean parpadeos o guiños en las distintas tiras. Para lograr una óptima integración en el ojo de los espectadores, se requiere al menos una frecuencia de refresco de 40Hz.

Además del posible riesgo de destellos indeseados en nuestro sistema, es necesario tener en cuenta que como consecuencia del refresco, aparentemente la luz será más tenue. Concretamente, su intensidad lumínica será 14 veces inferior a la máxima posible. Este efecto se debe a que, involuntariamente, se está implementando la técnica PWM para iluminar las tiras LED: se encuentran una catorceava parte del periodo de refresco encendidas y el resto del tiempo apagadas. Como ya se ha explicado anteriormente, esto se traduce en una intensidad lumínica considerablemente inferior. En razón de estas circunstancias, se va a optar por la siguiente propuesta.

La segunda alternativa que se propone es un circuito que cuenta con un mayor número de módulos WiFi para evitar las consecuencias negativas asociadas al empleo del demultiplexor.

Del mismo modo que en la primera estrategia, los módulos WiFi reciben la información de los cambios de colores inalámbricamente y deben controlar los drivers de las tiras en función de dicha información. Cada driver, como se ha descrito en el apartado anterior, cuenta con tres entradas de control que establecen la comunicación entre la tira LED y el microprocesador. Los módulos NodeMCU del proyecto cuentan solamente con 8 pines configurables como PWM y por tanto se necesitará un módulo por cada dos drivers (6 pines ocupados).

Elemento	Drivers	Módulos
Vidriera Delantera	16	8
Vidriera Trasera	15	8
Rosetón	10	5
<b>Total</b>	<b>72</b>	<b>37</b>

Figura 20: Unidades necesarias para todos los circuitos de alimentación del proyecto.

De acuerdo al diseño de los diferentes elementos, en la tabla se indican las cantidades de módulos y drivers necesarios para la instalación de los circuitos de control sin contar con los módulos maestros.

Atendiendo a su función, este circuito puede dividirse en dos secciones: alimentación de las tiras y control de las mismas.

### 3. Hardware

La alimentación de las tiras se realiza a 12V, a través de los drivers. Estos drivers se asociarán en paralelo y se alimentan desde la fuente. En cuanto al control de las tiras, se realiza desde los microprocesadores, que también necesitan ser alimentados. Su tensión nominal es de 5V que se extraerán de la fuente, como se describe anteriormente. Del mismo modo que los drivers, se alimentarán en paralelo.

Por último, el control parte del microprocesador y llega hasta los drivers, por lo que se necesitará un cable triple (uno para cada color).

El esquema del circuito de alimentación y control diseñado para el sistema es el siguiente:

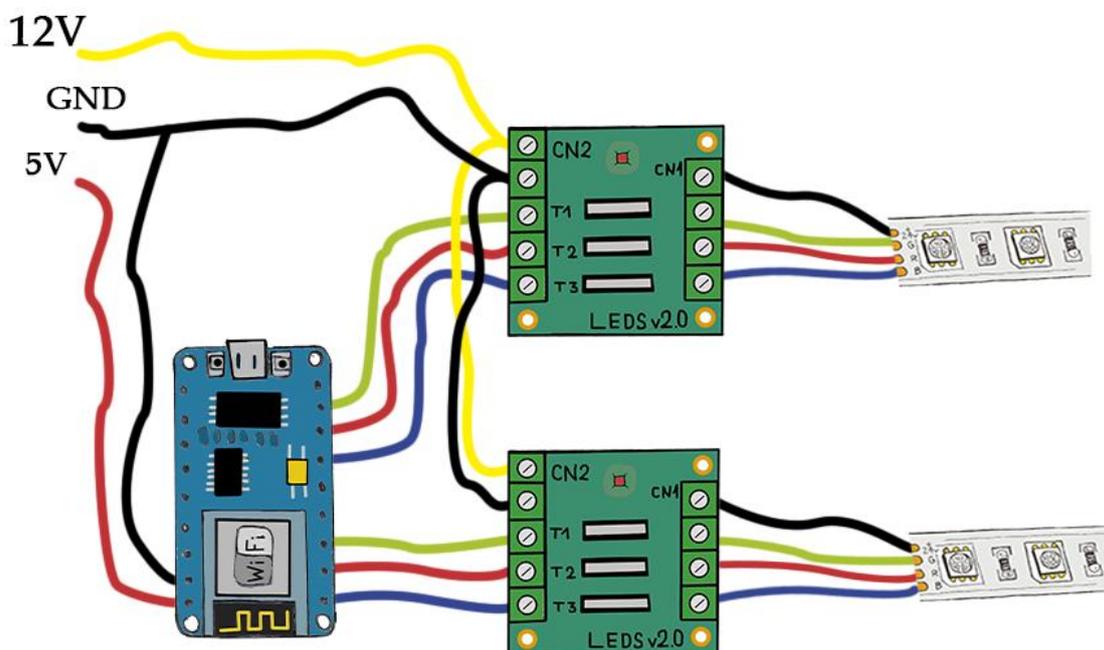


Figura 21: Esquema definitivo del circuito de la instalación

Sin embargo, el circuito adoptado presenta algunos inconvenientes respecto a la primera propuesta. Entre ellas se encuentra un aumento del coste debido a la necesidad de un mayor número de módulos WiFi. También produce un incremento del número de nodos de la red inalámbrica WiFi. De 5 nodos pasa a 37 sin incluir los módulos maestros, pero no hay ningún problema mientras no se alcancen los 250.

A pesar de estas desventajas, la solución adoptada permite la iluminación simultánea de todas las tiras LED, y con ello, mayor intensidad lumínica del sistema. Además reduce la complejidad del código que exigía el control del demultiplexor y por tanto ofrece la posibilidad de un desarrollo software más claro y directo.

## Capítulo 4

# **MONTAJE DE LA INSTALACIÓN**

Estructura y realización física, aspectos  
estéticos e instalación en el teatro

## 4.1 Estructuras y realización física

Una vez se han definido en profundidad las herramientas y componentes utilizados para el sistema de iluminación, es necesario describir los aspectos físicos del sistema y los requisitos de su construcción.

Aunque para el desarrollo del proyecto se trabaje simultáneamente el aspecto físico del sistema y su control software, se opta por describir previamente la realización física del conjunto a fin de facilitar la comprensión del código y los algoritmos utilizados.

En primer lugar hay que analizar las dimensiones y formas de los elementos escenográficos para poder elegir los diferentes materiales y fases de la construcción. Como ya se ha detallado anteriormente, se trata de construir 5 elementos escenográficos de grandes dimensiones: un rosetón, dos vidrieras traseras y dos vidrieras delanteras. La existencia de dos tipos diferentes de vidrieras se debe a las restricciones que el teatro y el espectáculo imponen. Estas especificaciones se abordan en apartados sucesivos.

Aunque la forma del rosetón es diferente a la de las vidrieras, su estructura es, en esencia, idéntica. Se quiere obtener secciones cerradas y aisladas unas de otras para conseguir iluminar cada superficie de manera independiente. En el caso del rosetón estas áreas son sectores circulares mientras que para las vidrieras, son cuadriláteros. Estas secciones deben corresponderse con los dibujos del material que actuará como difusor de la luz y también como acabado superficial para las vidrieras y el rosetón.

Para iluminar se aplican las tiras LED a lo largo de todo el perímetro de la sección aprovechando el adhesivo que incorporan.

Dejando al margen los aspectos estéticos, que se valorarán en el siguiente apartado, hay algunos requerimientos técnicos que las estructuras de nuestros elementos deben satisfacer:

- i. Se necesitan estructuras rígidas y resistentes ante posibles golpes producidos por choques con los elementos escenográficos móviles.
- ii. Debe tratarse de estructuras ligeras en relación a sus dimensiones para facilitar la instalación en el teatro.
- iii. Su material debe ofrecer una excelente conductividad térmica para disipar el calor generado por las tiras LED.

Al margen de estas solicitudes se tendrá en cuenta el precio del material seleccionado, dando prioridad al material más económico una vez se hayan satisfecho todos los requisitos. A fin de encontrar el material más adecuado, se plantea su construcción en madera, en plástico o en metal.

Atendiendo al tercer requisito, el que aborda el asunto de la disipación calorífica del sistema, el material más indicado parece ser un metal: hierro o aluminio. Los problemas que podrían surgir de una inadecuada evacuación del calor generado por los dispositivos electrónicos son diversos. Por un lado podría suceder un fallo eléctrico debido a una incorrecta temperatura de funcionamiento. Esta temperatura es crucial para algunos componentes electrónicos porque aumenta el consumo eléctrico del circuito.

Además de problemas en la instalación eléctrica, tanto el material difusor como las tapas traseras de las estructuras podrían sufrir daños o deterioros asociado a la exposición a temperaturas muy superiores a la ambiente. Incluso de un modo muy extremo, podría provocarse fuego por la ignición de alguno de estos materiales, que son considerablemente inflamables.

Una estructura metálica también satisface los otros dos requisitos expuestos. Si se utilizan perfiles metálicos tubulares para su construcción, la estructura resultante no será especialmente pesada mientras que si ofrecerá la rigidez propia de estos materiales.

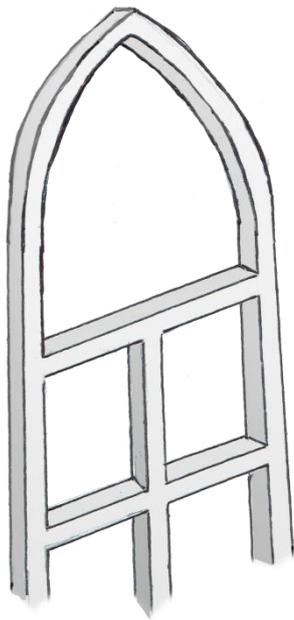


Figura 22: Extremo superior de una vidriera

Para concluir con la decisión del metal escogido, hay que elegir entre hierro o aluminio. Ambos metales se encuentran a la cabeza entre los más usados para estructuras en el ámbito industrial. En este caso, el metal escogido para construir las estructuras de la instalación es el hierro. La razón fundamental es la económica, el hierro puede tener un precio desde seis hasta diez veces inferior que el aluminio.

Escogido el hierro como material estructural de la instalación, es necesario diseñar las vidrieras y el rosón eligiendo los perfiles tubulares que los conformarán y dimensionándolos acorde al espacio disponible en el escenario.

Como aspectos constructivos relevantes, cabe destacar las dimensiones del tubo utilizado, 40x20mm para los segmentos horizontales y 40x40mm para el marco exterior. Ambos tubos cuentan con un espesor de 2mm. Estas secciones tubulares se unen formando los ángulos indicados en los planos utilizando soldaduras como método de unión.

En cuanto a las vidrieras traseras y delanteras, solamente aparece una dificultad en su construcción. La punta de la vidriera es un arco ojival de estilo gótico y se necesita deformar el tubo de hierro hasta conseguir ese aspecto. El rosón sin embargo, al tratarse de una figura redondeada, ofrece una problemática mucho mayor. Se acuerda con el técnico que lleva la construcción en hierro de los

#### 4. Montaje de la instalación

elementos, que en el caso del rosetón, los sectores circulares serán triángulos. Esto implica que la estructura resultante no es un círculo, sino un polígono regular de 16 lados, un hexadecágono. Esta decisión no tiene consecuencias sobre el aspecto estético del elemento final porque se recuerda que la única parte expuesta de todo el conjunto será una imagen que sí respetará el aspecto circular tradicional de los rosetones.

Una vez que se tienen las estructuras, se debe escoger el material de las tapas traseras del conjunto. La función de esta tapa trasera es doble. En primer lugar bloqueará la salida de la luz de los LED actuando como reflector. De este modo, se consigue una distribución mucho más homogénea de la misma. Para que sea lo más homogénea posible, las tapas traseras deben ser blancas y, si es posible, estar un poco satinadas. El material escogido es madera tipo MDF con una de las caras en blanco. Se valora el uso de una lámina fina de metal para la parte trasera, ya que permitiría una buena disipación calorífica. A pesar de esa ventaja, se descarta por razones económicas y también porque la manipulación de este material es más complicada que en madera.

Sin embargo, aunque actuar como reflector es importante, otra funcionalidad de estas tapas traseras es albergar toda la circuitería: drivers, módulos WiFi y todo los cables necesarios.

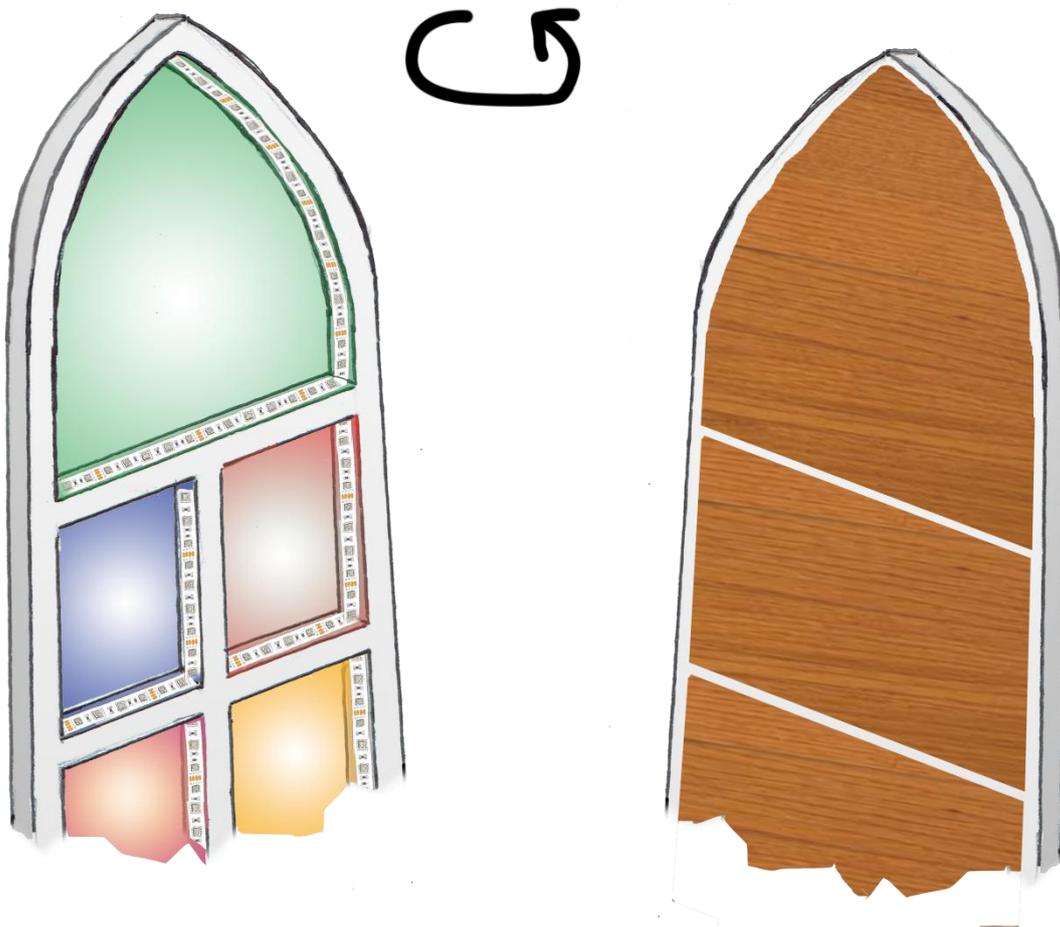


Figura 23: Ilustración esquemática de una vidriera con las tiras LED y por detrás, con las tapas de madera.

Aprovechando la superficie trasera de la estructura para soportar todos los componentes electrónicos como si se tratase de una enorme placa se consigue mantener orden y claridad en la instalación. De este modo, en caso de fallo de cualquier tipo, el error puede detectarse rápidamente. Además se dejan accesibles todas las conexiones de tal modo que reponer un componente deteriorado es realmente sencillo.

La distribución del circuito sobre la tapa trasera es diferente en el rosetón y en las vidrieras por razones geométricas, pero conceptualmente se trata de un esquema muy similar:

- i. La alimentación de las tiras LED se hace a través de los drivers, que se encuentran repartidos cada uno detrás del panel que ilumina.
- ii. El cableado que proporciona los 12 V en el circuito va uniendo los drivers de dos en dos. De este modo, la conexión en paralelo requiere una cantidad inferior de cables, lo que implica más orden y claridad en el circuito así como una reducción del presupuesto.
- iii. La alimentación de los módulos WiFi a 5v es similar, pero ocupan la vertical que pasa por el centro de la vidriera. Así, cada módulo se encuentra posicionado a la altura de los paneles que maneja.
- iv. El cableado de control une los módulos WiFi con los drivers a través de 3 cables horizontales.

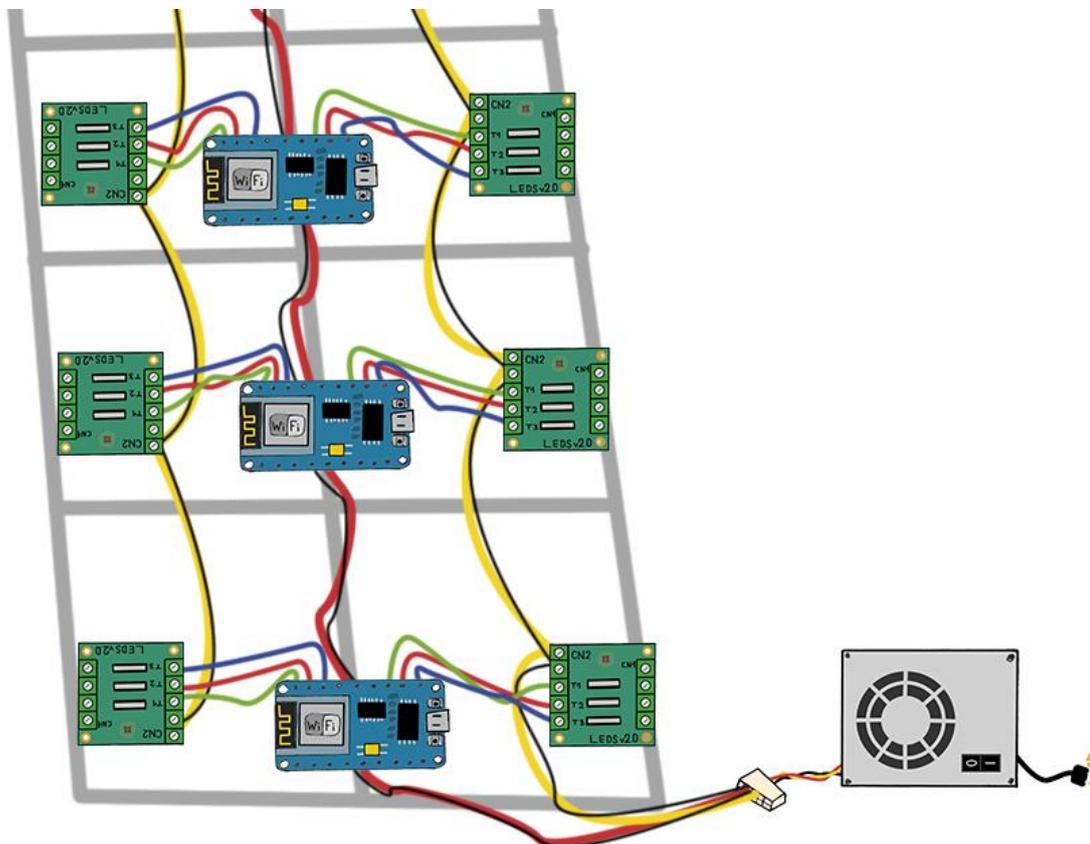


Figura 24: Ilustración esquemática de los circuitos y su disposición en la parte trasera de una vidriera

## 4.2 Aspectos estéticos

Aunque se trate de un asunto secundario en un proyecto de Electrónica y Automática, la imagen de los productos desarrollados es crucial en cualquier campo de la Ingeniería. De hecho, este proyecto trata de un producto realmente visual. Por ello, se ha decidido dedicar un breve apartado a los aspectos estéticos de la instalación.

Como se ha comentado en apartados anteriores, se necesita un material difusor para homogeneizar la intensidad lumínica de cada panel. Además, debe transformar las estructuras metálicas cableadas en cuatro vidrieras y un rosetón de estilo gótico.

Para escoger el material que realizase estas funciones se evalúan diferentes alternativas. En primer lugar se descartó el empleo de cristal o metacrilato rígido por motivos económicos. Por otra parte, estos materiales no son capaces de repartir la luz para que no se aprecie que los LED se sitúan bordeando cada panel. Además, en el caso del cristal se suma la desventaja de su fragilidad. Al fin y al cabo, cualquier elemento escenográfico se encuentra permanentemente bajo el riesgo de ser golpeado y entonces, por motivos de seguridad, no se trata de una buena opción.

Se opta entonces por un material plástico que no sea transparente y así pueda retener la luz en su superficie. También se decide acudir a una imprenta capaz de realizar trabajos de gran formato para la "piel" de las estructuras. Imprimir sobre un plástico flexible que haga las veces de difusor permite instalar sobre las estructuras las imágenes que se quieran con un acabado óptimo. La preparación de estas imágenes es llevada a cabo por un diseñador gráfico externo al que se le han facilitado las dimensiones de las estructuras y algunas directrices sobre la estética adoptada.

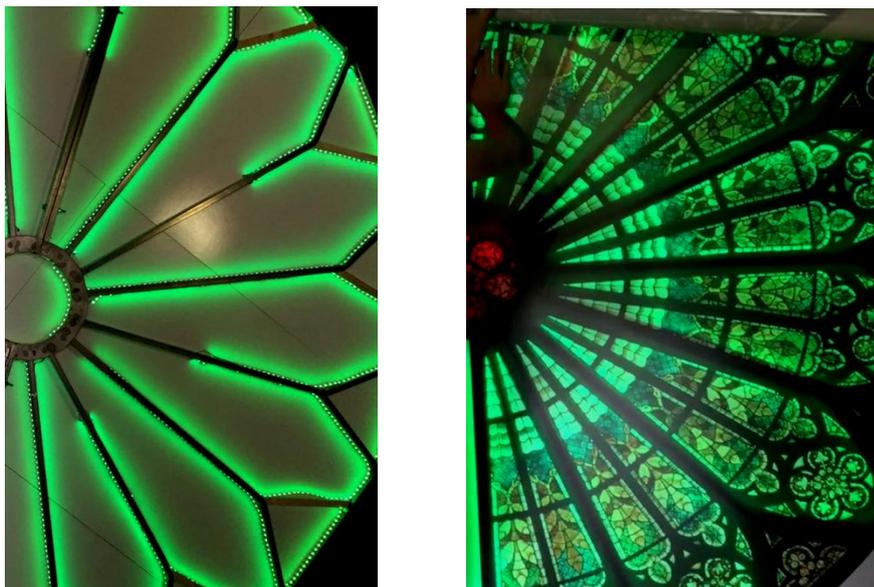


Figura 25: Imagen para comparar la estructura del rosetón sin el papel difusor y con él

## 4.3 Instalación en el teatro

Cuando ya se han preparado las estructuras incluyendo los circuitos eléctricos que necesitan y su plástico difusor, se procede a la instalación definitiva en el escenario del teatro. La finalidad de este proceso es situar de manera permanente los elementos escenográficos fijos y también aquellos que cuenten con algún sistema de tracción. Una vez finalizada la instalación se procede a realizar los ensayos y pruebas pertinentes para asegurar el correcto funcionamiento de todos los elementos.

Entre todos los elementos diseñados y construidos, hay tres fijos y dos que contarán con un sistema que permitirá su movimiento dentro del escenario. La razón que motiva la necesidad de instalar las vidrieras delanteras con dicho sistema es que ocupan un lugar en el escenario que interrumpe el desarrollo del espectáculo durante las escenas en las que no forman parte del decorado.

Por tanto, el rosetón y las vidrieras traseras se fijarán a la pared que está en el fondo del escenario aplicando tornillos de gran calibre. El proceso de instalación y anclaje en el teatro lo realiza el mismo técnico que construyó las estructuras de los elementos en metal. La precisión y robustez con la que se realice esta actividad es primordial para garantizar la seguridad de todo el equipo de actores y técnicos durante el desarrollo de las funciones. A fin de evitar posibles accidentes se aseguran todos los anclajes sobredimensionando las uniones roscadas utilizadas así como el número de las mismas.

De acuerdo a las condiciones impuestas por las dimensiones del escenario, las vidrieras traseras cuentan con una altura de 4 metros desde su base hasta la punta y sin embargo, su espesor no es más de 20mm. Si no se fijan adecuadamente, estas estructuras serían extremadamente inestables. Esto se debe a que no se apoyan contra la pared del fondo, a diferencia del rosetón. Los motivos por los que las vidrieras traseras no se apoyan contra el fondo del escenario son dos: uno de carácter técnico y otro estético.

La razón técnica que motiva la decisión de adelantar las estructuras respecto de la pared unos 40cm es facilitar el acceso al cableado y componentes electrónicos. Durante las pruebas de iluminación es necesario tener un acceso cómodo y rápido a los módulos WiFi para reprogramarlos tan rápido como sea posible. Esto será imprescindible para modificar algún aspecto del código o ajustar las tonalidades e intensidad de los colores.

En cuanto al aspecto estético, alejando las vidrieras mínimamente desde la pared trasera, donde se encuentra instalado el rosetón, se consigue un mayor efecto de profundidad y perspectiva. Además, para reforzar aún más este efecto, no se situarán paralelas a la pared, sino con una ligera inclinación.

El sistema mecánico de soporte y anclaje de las vidrieras traseras, por tanto, son unos tubos de hierro de sección cuadrada normales a la pared fijados a la misma a través de uniones roscadas.

#### 4. Montaje de la instalación

El rosetón, sin embargo, se encuentra anclado a la pared del fondo del escenario con su superficie trasera en contacto con la misma. Se realiza de esta manera porque el rosetón no tiene un apoyo en el suelo y si se alejase de la pared, el sistema de fijación podría flectarse generando un riesgo que no se puede correr.

Las vidrieras traseras y el rosetón pueden ocultarse utilizando un telón negro que se encuentra por delante de ellos. Sin embargo, las dos vidrieras delanteras necesitan un sistema de fijación que las permita desplazarse lateralmente para ocultarse en las escenas donde no forman parte del decorado.

Para ello se instala un sistema de raíles en el techo del escenario. El extremo superior de la estructura de las vidrieras delanteras cuenta con unas ruedas ajustables que encajan en las guías. De este modo las ruedas son capaces de recorrer toda la longitud del raíl.

Al tratarse de un proyecto que se implementa en un escenario real, las dimensiones y aspectos constructivos que se diseñan deben ser contrastadas en detalle con el espacio escénico para determinar su viabilidad. Por ello, las estructuras construidas se ajustan a la perfección al teatro, lo que concede al conjunto un aspecto cuidado y equilibrado sin grandes espacios vacíos ni agolpamiento de tantos elementos. Además se ha procurado reforzar y exagerar la perspectiva de los elementos y su posición para transmitir la sensación al espectador de encontrarse dentro de una gran iglesia.

La figura es una montaje fotográfico que muestra el resultado que se pretende obtener, aunque no incluye las dos vidrieras delanteras. En la imagen se puede apreciar también una figura de la Virgen que formará parte de la decoración junto a los elementos escenográficos proyectados.



Figura 26: Fotomontaje del escenario que representa la instalación del sistema en el teatro

Capítulo 5

# **RED DE COMUNICACIÓN INALÁMBRICA**

Características y establecimiento de la red

## 5.1 Características de la red

El empleo de una red de comunicación inalámbrica supone una reducción del cableado que, consecuentemente, facilita la instalación disminuyendo el tiempo necesario. Además la convierte en una red mucho más flexible. La consecuencia directa de este incremento de la versatilidad es la posibilidad de añadir o reducir el número de nodos presentes en la red sin ninguna dificultad.

Sin embargo, también existen riesgos asociados al uso de esta tecnología WiFi inalámbrica. [22] En primer lugar, se pueden dar fallos en la conexión, mientras que en el empleo de cableado esto no es tan común. Para controlar este factor se creará una función para los módulos WiFi NodeMCU capaz de identificar la potencia de la red. Estos fallos de conectividad pueden propiciarse por las interferencias de otras ondas y las pérdidas de señal que el ambiente pueda acarrear. Los obstáculos físicos también son un problema para la transmisión de este tipo de ondas.

Un aspecto fundamental, y que se abordará en otro apartado, es la seguridad de la red. La seguridad de una red es crucial en cualquier aplicación si se quiere preservar la privacidad de la comunicación. Para esta aplicación se necesita transmitir información muy precisa, pero su privacidad no es tan imprescindible. No se están transmitiendo datos que comprometan a ninguna persona pero la entrada de algún usuario extra a la red podría generar interferencias. Por supuesto, si el usuario accediese a la Interfaz Gráfica de Control, la aplicación podría ser hackeada perturbando el desarrollo normal de la función.

Al margen de estos factores, se procede a estudiar la topología de la red. [23] La topología de red o forma lógica se define como la cadena de comunicación que los nodos que conforman una red usan para comunicarse. Es la distribución geométrica de los nodos conectados, aunque no tiene nada que ver con su ubicación espacial. Las topologías más recurrentes son la red bus, la red en anillo y la red estrella.

La topología de una red bus se caracteriza por tener un único canal de comunicaciones (denominado bus, troncal o backbone) al cual se conectan diferentes dispositivos. Tiene todos sus nodos conectados directamente a un enlace y no tiene ninguna otra conexión entre sí. Esta topología es característica de redes cableadas y entre sus ventajas se encuentran la facilidad de implementación y su simplicidad. Por otra parte, sus desventajas son la limitación de la longitud del canal y las altas pérdidas en la transmisión ocasionadas por las colisiones entre mensajes.

La red en anillo es una topología en la que cada estación está conectada a la siguiente y la última está conectada a la primera. Cada estación tiene un receptor y un transmisor que hace la función de repetidor pasando la señal a la siguiente estación. Las desventajas de esta topología es la degradación de la

transmisión conforme la red va creciendo. Además, en caso de fallo en un nodo, la red se daña completamente.

Por último, la alternativa más rápida y eficaz es la topología en estrella. Permite la fácil detección de nodos con fallos así como variar la extensión de la red. Todos los nodos de la red se encuentran conectados a un punto central. De este modo, todas las comunicaciones se han de llevar a cabo necesariamente a través de éste. Este esquema ofrece, además, una mayor seguridad y robustez a la red.

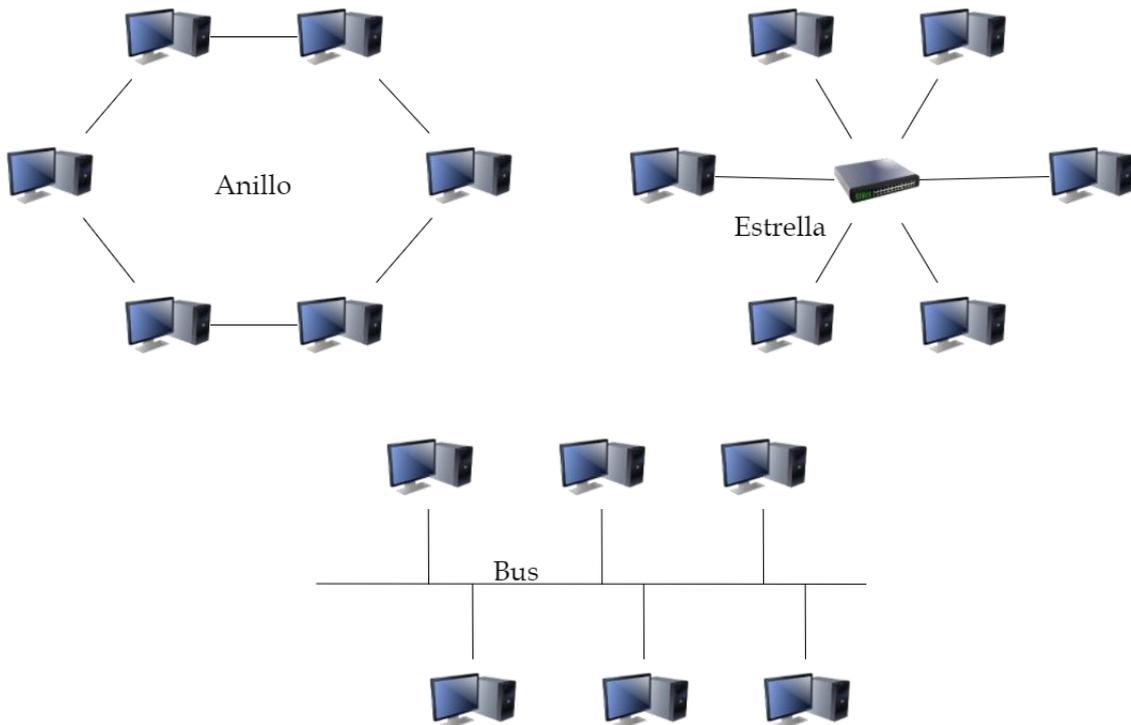


Figura 27: Topologías de Red, de izquierda a derecha: anillo, estrella y bus

Por tanto, es necesario definir el punto central de la red, también conocido como Punto de Acceso. Es el responsable de crear la red que va a articular todas las comunicaciones del sistema. Además, debe dar un nombre a la red para que todos los nodos sean capaz de encontrarla y conectarse.

Se conoce como *SSID*, *Service Set Identifier*, al nombre incluido en los paquetes de una red inalámbrica de tal manera que quedan identificados como parte de esa red. El nombre adoptado por la red de la instalación es **PHILADELPHIA**. Hace referencia a la ciudad donde transcurre la historia del espectáculo. Este SSID se configura desde el Punto de Acceso a través de una Web que genera el mismo en la primera dirección IP de la red.

Para hacer un sistema de comunicación universal, se necesita un método de identificación de los nodos aceptado globalmente. La dirección IP es el

## 5. Red de Comunicación Inalámbrica

identificador de cada nodo dentro de una red. La IP de un nodo lo define entonces unívocamente.

[24] Las direcciones IP se representan como cuatro enteros decimales separados por puntos, donde cada entero da el valor de un octeto de la dirección. Así, por ejemplo, la dirección de 32 bits:

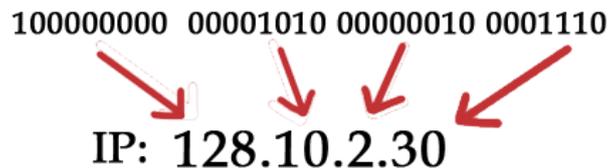


Figura 28: Explicación gráfica sobre la formación de una dirección IP de 32 bits

El conocimiento y manejo de las direcciones IP de los nodos con los que se cuenta es fundamental si se pretende discriminar el envío de mensajes. El protocolo de transmisión de información que se utiliza es UDP. Los paquetes que se envían son mensajes de texto que además necesitan información relativa al destinatario del mismo.

Cada dirección IP se forma con un par de identificadores: identificador de red e identificador del host. Según la cantidad de bits de la IP dedicados a definir la red y el nodo, las direcciones IP se pueden clasificar en clases. En la práctica hay tres clases distintas de direcciones (clases A, B y C):

Clase A (1.0.0.0 a 126.255.255.255)				
0	Redid (7 bits)		Hostid (24 bits)	

Clase B (128.0.0.0 a 191.255.255.255)			
1	0	Redid (14 bits)	Hostid (16 bits)

Clase C (192.0.0.0 a 223.255.255.255)				
1	1	0	Redid (21 bits)	Hostid (8 bits)

Figura 29: Clases IP

Por tanto, dada una dirección IP, se puede terminar su clase a partir de los tres bits de orden más alto. Las direcciones de Clase C se usan para redes que tienen hasta 254 hosts o menos, utilizan 21 bits para Redid y solamente 8 para Hostid. También se reservan ciertas direcciones para determinados usos, entre los cuales están las direcciones para redes privadas o intranets.

Las direcciones de las intranets no se usan como globales en Internet. Se reservan para la creación de redes no conectadas Internet. La red de nuestro sistema de comunicaciones inalámbricas se adhiere a esta clasificación. No se requiere conexión a Internet para el funcionamiento de las instalaciones y cuenta con un número considerablemente menor que 254. Por tanto, el

direccionamiento IP será de clase C e intranet: direcciones de la 192.168.0.0 a la 192.168.255.255.

La asignación de cada una de estas a los módulos WiFi que la soliciten se produce en el momento de conexión del nodo a la red. Esto quiere decir que las direcciones IP son dinámicas y dependen, de algún modo, del momento de conexión a la red. Una vez que el dispositivo se desconecta, esa dirección IP queda liberada y, por tanto, se puede asignar a otro nodo.

La ventaja fundamental de este sistema de asignación de direcciones IP, que realiza el creador de la red, es una mayor flexibilidad y sencillez a la hora de hacer crecer o reducir la extensión de la red.

Sin embargo, dentro de las direcciones IP disponibles dentro de una red del tipo que se va a crear, hay algunas direcciones reservadas:

<b>Dirección IP:</b>	<b>Función:</b>
192.168.1.0	Se utiliza para designar la propia red
192.168.1.1	Corresponde al Punto de Acceso <i>AP</i>
192.168.1.255	Se utiliza para hacer un broadcast

La dirección broadcast se refiere a todos los nodos conectados a la red. El beneficio principal de la existencia de esta dirección IP es que permite enviar un paquete a todos los módulos que pertenecen a la red. Esta estrategia de comunicación es frecuentemente utilizada en redes con grandes cantidades de nodos receptores. Para conseguir una buena sincronización entre todos los actuadores del sistema, se empleará la dirección broadcast para enviar a todos los módulos un único mensaje de caracteres. Después, cada nodo extraerá la información del mensaje que le corresponda para actuar en consecuencia.

Este paquete se conoce como datagrama y es la unidad básica de transmisión de datos en una red tipo WiFi. Un datagrama, del mismo modo que en las redes físicas, se divide en encabezamiento y campo de datos. El encabezamiento contiene las direcciones IP de la fuente y el destino.

El protocolo de comunicación utilizado, como se explica en el apartado de herramientas, es UDP. Este protocolo permite identificar rápidamente al destinatario final de un datagrama. El principal beneficio de usar este protocolo es su fluidez. Esto es debido a que no cuenta con acuse de recibo, a diferencia del protocolo TCP. Es decir, el nodo emisor envía su mensaje a una dirección IP concreta, pero no recibe ninguna confirmación de llegada correcta. De este modo se establece un tráfico de paquetes en cadena abierta, sin realimentación.

El protocolo UDP incluye los denominados puertos. Los puertos identifican el último destino dentro del módulo WiFi, el programa de aplicación que usa dicho puerto. Se identifican con un número entero. Los puertos de los módulos

## 5. Red de Comunicación Inalámbrica

empleados en la red del sistema es el número 2390, y debe ser activado en la programación de los microprocesadores.

Una vez se ha preparado el mensaje, el protocolo UDP entrega al servicio IP el segmento que contiene los datos, y una cabecera que no se transmite, pero que permite al protocolo IP generar correctamente los datagramas.

El formato de un paquete UDP es el representado en la figura:

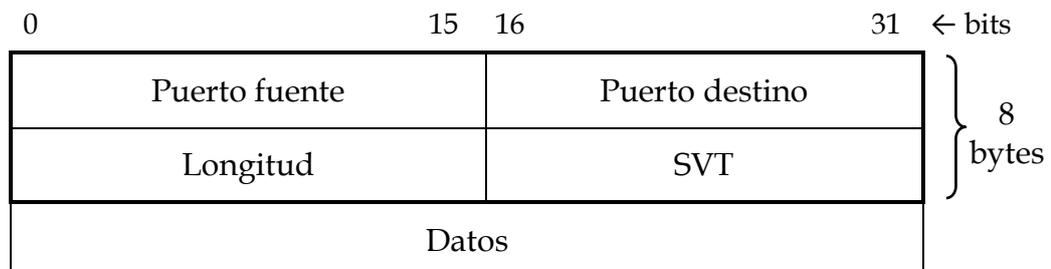


Figura 30: Formato de la cabecera del protocolo UDP

- i. Los puertos fuente y destino son valores de 16 bits correspondientes a los puerto de transporte.
- ii. La longitud es el número total de bytes que contiene el paquete UDP original incluyendo la cabecera, antes de ser fragmentado en paquetes IP.
- iii. SVT significa suma de verificación, aplicada a la cabecera y datos UDP.
- iv. Los Datos son el mensaje de texto que se pretende enviar.

Cuando se han definido los principios de la red que va a soportar las comunicaciones del sistema y los protocolos empleados, se procede a explicar el establecimiento de la misma y los aspectos de seguridad.

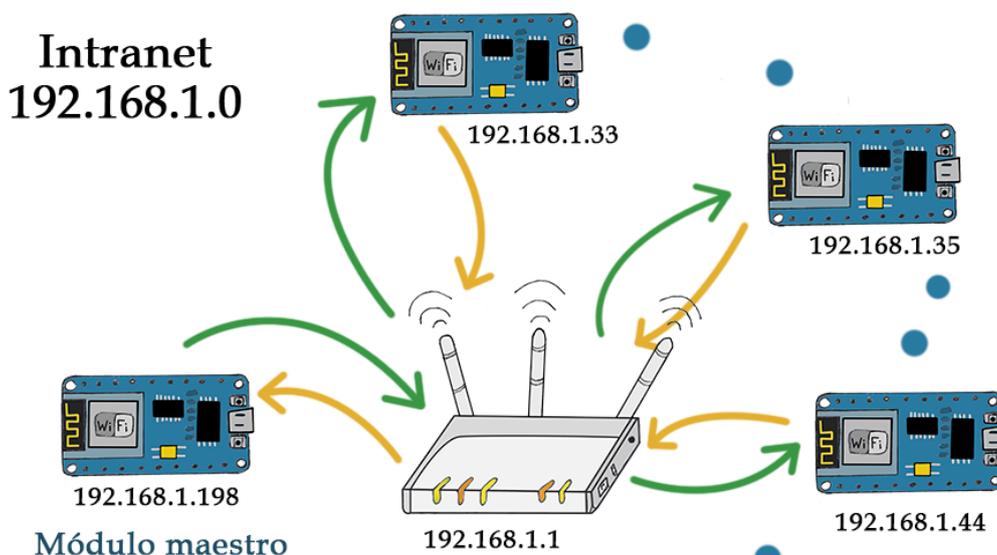


Figura 31: Modelo de la red 192.168.1.0

## 5.2 Establecimiento de la intranet

Para crear la red diseñada destinada a la transmisión de mensajes relativos al cambio de colores en los LED, es necesario configurar el Punto de Acceso correctamente. El objetivo de este apartado es describir el procedimiento necesario para establecer una red de área local, LAN.

La intranet establecida soportará un Servidor Web para la Interfaz Gráfica de Control del sistema además de la transmisión de paquetes UDP. Sin embargo, es imprescindible modificar algunos parámetros de la red local como por ejemplo, la seguridad de la misma.

En primer lugar hay que escoger el router WiFi que actuará como Punto de Acceso de la red de área local. En el mercado hay una gran variedad de dispositivos de este tipo y su labor es encaminar datagramas entre diferentes direcciones IP. En este proyecto se va a utilizar un router reconfigurándolo para que funcione como creador de la red sin acceso a internet, es decir, una red local.

El procedimiento que hay que seguir para configurar adecuadamente esta red cuenta con los siguientes pasos:

- i. Encender el router y acceder a su interfaz de configuración a través de un navegador de Internet. Para ello, hay que introducir en la barra de direcciones una de estas 3 direcciones IP, ya que depende del fabricante: 192.168.1.1 - 192.168.2.1 - 192.168.0.1
- ii. Introducir los datos que solicita, habitualmente usuario y contraseña. Si no se conocen estos parámetros se puede resetear el router e introducir los valores iniciales contactando con el fabricante, si estos también se desconocen.
- iii. Los parámetros que se necesita configurar son:

IP address:	192.168.1.1
Subnet Mask:	255.255.255.0
Servidor DHCP:	activado
Start IP address:	192.168.1.30
End IP address:	192.168.1.255
- iv. En algunos router es también imprescindible abrir los puertos para activar las comunicaciones a través del protocolo UDP.

La asignación de direcciones IP de una red pueden ser estáticas o dinámicas. La dirección IP es estática cuando la asignamos nosotros manualmente en el dispositivo (ordenador, disco de red, etc). Las direcciones dinámicas, sin embargo, las asigna el servidor DHCP de forma transparente para el usuario.

## 5. Red de Comunicación Inalámbrica

Esto quiere decir que el dispositivo una vez conectado al router adquiere una dirección IP automáticamente, y en principio, queda configurado y listo para utilizar la conexión inalámbrica.

Aunque el concepto de asignación dinámica es realmente cómodo, resulta muy útil tener reservadas unas cuantas direcciones IP estáticas, por si fuese necesario añadir un nodo a la red que requiriese este tipo de direcciones permanentes. Para ello, el valor que se ha introducido en la configuración para el campo Start IP address es el 192.168.1.30. Se han reservado todas las direcciones entre la 2 y la 29 para su asignación manual, mientras que el resto, de la 30 a la 254, las asignará el servidor DHCP por orden de entrada a la red establecida.

Además de todos estos valores, durante esta configuración se establecen el nombre de la red, SSID, y la contraseña de acceso. El SSID es obligatorio para permitir a los nodos encontrar la red. SSID es una cadena de caracteres alfanuméricos que habitualmente viene preestablecida desde fábrica, pero que es configurable. Sin embargo, la contraseña de acceso es opcional.

La clave de acceso de un red WiFi tiene como objetivo fundamental limitar la conexión a la red a dispositivos no autorizados, si se considera que la autorización viene dada por el conocimiento de la contraseña. Para cualquier aplicación que utilice esta tecnología inalámbrica, proveer a la red de las medidas de protección necesarias es realmente importante. Un acceso no deseado a una red de área local podría como mínimo interrumpir el correcto funcionamiento de la actividad que se esté desarrollando. Además, el usuario que ha entrado podría tener acceso a información privada o incluso hacerse con el control de la aplicación si cuenta con los conocimientos necesarios para ello. Por no hablar de la notable reducción del ancho de banda que implica una disminución considerable de la velocidad de las comunicaciones.

Si la seguridad es vital en cualquier tipo de comunicación inalámbrica, lo es aún más en el caso de las redes WiFi. Esto es debido principalmente al hecho de que en la actualidad cualquiera cuenta con un dispositivo capaz de acceder a estas redes: su Smartphone. Esta amenaza puede reforzarse en este proyecto ya que el producto y, en concreto, la red inalámbrica estará expuesta al presunto intento de conexión de más de 400 teléfonos móviles.

Dicho esto, resulta obvia la necesidad de aplicar un método eficaz para prevenir estas intrusiones indeseadas. Para ello, se cuenta con una amplia variedad de posibilidades que protegen nuestra red y dispositivos. [25] Estos protocolos pueden bloquear el acceso a un tipo determinado de dispositivos o crear encriptaciones de la contraseña para que no pueda ser obtenida utilizando ningún software dedicado al hackeo de sistemas.

Entre todas las posibilidades, resulta muy recomendable seguir las siguientes indicaciones:

- i. Asignar el sistema de seguridad más avanzado: WPA2 con un sistema de cifrado o encriptación AES.

- ii. Cambiar la contraseña por defecto. Siendo recomendable el empleo de al menos 12 caracteres con mayúsculas, minúsculas, números y símbolos.
- iii. Cambiar el nombre de la WiFi (SSID).
- iv. Modificar la contraseña para cambiar la configuración del router, de este modo se evita que un intruso pueda configurarlo a su antojo.
- v. Habilitar la restricción MAC (o dirección física). Implica la restricción del acceso a la red tan solo a aquellos equipos con una dirección MAC concreta. MAC es un identificador único de cada dispositivo de red y se pueden añadir al router como dispositivo seguro.

Capítulo 6

# **DESARROLLO SOFTWARE**

Introducción, Diseño general e  
implementación

## 6.1 Introducción al desarrollo del software

Durante los capítulos previos se ha abordado el diseño estructural de la instalación, el diseño de los circuitos y el establecimiento de la red para las comunicaciones inalámbricas. Una vez se han resuelto todas estas cuestiones, se está en condiciones de afrontar el diseño y desarrollo del software que regirá el funcionamiento del sistema de iluminación RGB y su sincronización con el espectáculo.

Sin embargo, la creación de software es una actividad que requiere tiempo y dedicación tanto en sus primeros estadios de desarrollo como en las últimas pruebas. Por ello, el desarrollo del mismo debe avanzar simultáneo y en paralelo al resto de actividades, como se refleja en el diagrama Gantt del proyecto.

La estrategia que se define para la creación del software cuenta con varias fases:

- i. Estudio de las herramientas de desarrollo a fin de familiarizarse con las mismas, tanto el lenguaje como las librerías. Esta fase incluye también la preparación de la IDE Arduino en el ordenador desarrollador y la instalación de los drivers pertinentes para trabajar con los módulos WiFi NodeMCU.
- ii. Diseño de un programa general que articule las posteriores líneas de desarrollo y que sirva de referencia durante la creación del código llevando el control de variables y funciones.
- iii. Primeras pruebas en una maqueta simplificada de la instalación donde se compruebe el correcto funcionamiento de las comunicaciones y la circuitería diseñada. Se plantea un sistema reducido que cuente solamente con dos tiras LED RGB que controlar, los dos drivers correspondientes, un módulo esclavo y un módulo maestro. A través de estas pruebas se consigue depurar el código desarrollado hasta entonces. Esto es fundamental para un posterior desarrollo software más completo y complejo.
- iv. Ampliación del sistema trasladando las pruebas a una vidriera completa. Esto supone el comienzo de la programación de rutinas de iluminación definitivas ya que se controlan las 15 tiras LED RGB simultáneamente de un elemento completo.
- v. Inclusión del rosetón y el resto de vidrieras a las pruebas para simular el funcionamiento final de todo el conjunto. Preferiblemente esta fase será realizada en las inmediaciones del teatro para poder probar y ajustar los distintos efectos y colores teniendo una visión global y definitiva de la instalación.

## 6.2 Puesta a punto de las herramientas

De acuerdo con el capítulo de herramientas, el desarrollo del software se lleva a cabo a través de la IDE Arduino. Sin embargo, antes de comenzar con el código es necesario instalar los drivers pertinentes para permitir la comunicación serial entre el ordenador desarrollador y los módulos WiFi NodeMCU.

Para ello, hay que acudir al Panel de Control de la máquina y asociar los drivers a los puertos USB del mismo. Estos drivers pueden encontrarse disponibles de manera gratuita en la Web. Se recuerda que Arduino es una plataforma libre de desarrollo.

No obstante, para estar en disposición de compilar y cargar programas en las placas es imprescindible familiarizarse con las librerías y el lenguaje propio de Arduino. Una de las ventajas de utilizar este software libre es el acceso a una ingente cantidad de librerías que facilitan el desarrollo del código. [26]La propia Web de Arduino ofrece una sección de referencias donde se explican y presentan diferentes aspectos sintácticos del lenguaje propio de Arduino así como una selección de librerías oficiales.

[27]Una librería es una colección de ficheros objeto. Estos ficheros objeto contienen una amplia variedad de funciones y clases que pueden ser utilizadas una vez se ha incluido en el código la librería a la que pertenecen. De este modo el compilador será capaz de unir el código desarrollado por el programador con el código de las funciones implementadas desde librerías externas.

De cara al código y funcionalidades que requiere este proyecto, hay dos librerías que son necesarias para su desarrollo: la librería *ESP8266WiFi* y a librería *WiFiUdp*.

La librería *ESP8266WiFi* debe ser incluida porque permite controlar desde el microprocesador las diferentes funcionalidades del módulo WiFi ESP8266, el utilizado en este proyecto. Esta librería presenta las clases *WiFiClient* y *WiFiServer* entre otras, necesarias para utilizar el módulo WiFi como servidor Web, soporte de la Interfaz Gráfica de Control.

Por otra parte, la librería *WiFiUdp* habilita a los nodos de la red para enviar y recibir datagramas. Permite configurar los puertos locales y preparar los paquetes UDP que se deseen enviar construyendo la cabecera de un modo muy sencillo y transparente para el usuario. Entre los métodos de la clase *Udp* que más se utilizan se encuentran los relativos a la construcción de los mensajes: *Udp.beginPacket(IP, localPort)* y *Udp.write(mensaje)*.

La inclusión de estas librerías se realiza en la cabecera del código fuente utilizando las sentencias :

```
#include <ESP8266WiFi.h>
#include <WiFiUdp.h>
```

El lenguaje de programación propio de Arduino es relativamente similar al lenguaje de alto nivel C. Sin embargo, se presentan algunas diferencias entre ambos lenguajes. Uno de los aspectos distintivos es la existencia de funciones que surgen desde la necesidad de gestionar las entradas y salidas de los pines de las placas. Estas funciones permiten tanto leer información en los pines, como escribir el valor de las salidas cuando estos están configurados como tal.

Para el desarrollo de este proyecto cabe destacar la función *analogWrite( pin , value )*. Como ya se ha explicado en el apartado de herramientas, esta función concede la posibilidad de aplicar la técnica PWM para iluminar los LED. Sin embargo, esta función solo puede ser aplicada en los pines preparados para ello. Esta limitación debe ser tenida en cuenta a la hora de configurar los pines. Sólo son capaces de implementar la técnica PWM si delante de su numeración en la placa cuentan con una virgulilla: ~ .

El argumento *value*, por su parte, representa el ciclo de trabajo de la onda cuadrada y su valor debe estar comprendido entre 0 (siempre apagado) y 255 (siempre encendido).

Además, Arduino cuenta con funciones realmente útiles cuando se pretende construir una aplicación que necesita controlar el tiempo. Una vez el conjunto esté en funcionamiento, la sincronización de los efectos luminosos se llevará a cabo a través del control temporal de las transiciones. Estas funciones, por ejemplo, son: *delay( time )* o *millis( )*. La función *delay* bloquea el progreso de las rutinas programadas durante los milisegundos que indique su argumento *time*. Por otra parte, *millis( )* devuelve en formato *unsigned long* el número de milisegundos desde el comienzo del programa. Este número vuelve a cero una vez que se produce su desbordamiento, aproximadamente 50 días.

Dentro de toda la programación del sistema, cabe destacar un aspecto sintáctico de mayor complejidad: la aritmética de punteros.

El empleo de punteros y la gestión de sus posiciones de memoria surge de la necesidad de realizar cambios en los elementos de una cadena de caracteres dentro de funciones creadas para ello. Si estos vectores no se pasan como parámetros de la función en cuestión, no se tendrá acceso a su posición de memoria y por tanto será imposible modificar los valores del mismo.

Entonces el prototipo de la función debe contar al menos con un parámetro declarado de esta manera:

```
void Change_First_Element (char c, char *string ) {  
    string [ 0 ] = c ;    }
```

Así, una vez se retorna de la función, el primer elemento de la cadena que se deseaba modificar por el carácter introducido, efectivamente se modifica. El símbolo \* hace referencia a un puntero que señala a la dirección de memoria

## 6. Desarrollo Software

del primer elemento de la cadena y, por tanto, debe ser introducido como parámetro del siguiente modo al llamar a la función:

```
Change_First_Element ( 'A', &cadena[ 0 ] );
```

La importancia de una buena práctica en aritmética de punteros para el manejo de vectores y cadenas de caracteres viene dada por la necesidad de generar los mensajes necesarios para el control de la iluminación.

Además de estas consideraciones relativas a la sintaxis y las librerías que se van a implementar, es interesante definir las variables características del código que se pretende desarrollar:

<b>Tipo de dato</b>	<b>Variable</b>	<b>Función</b>
STRING	mensaje[28]	Almacenar el contenido de los datos del paquete UDP que se desea enviar.
STRING	colorV[16]	Contiene la combinación de colores de los paneles de Vidrieras codificados como caracteres mayúsculos.
STRING	colorR[10]	Análoga a colorV[ ], salvo que orientada al Rosetón.
INT	localPort	Habilita los puertos del protocolo UDP.
INT	temp	Variable auxiliar que controla los tiempos de ejecución de las distintas rutinas.
STRING	Broadcast[13]	Contiene la dirección IP que permite hacer broadcast: 192.168.1.255.
CONST CHAR*	idWiFi	Almacena la SSID de la red creada para las comunicaciones: PHILADELPHIA.
CONST CHAR*	password	Guarda la contraseña de conexión a la red de área local PHILADELPHIA.
INT	color[3]	Representa un color genérico. Almacena los valores correspondientes al PWM en los LED. Por ejemplo: azul [ 0 , 0 , 255 ].

## 6.3 Diseño del Software. Flujograma.

Una vez establecidas las bases de la programación que se va a desarrollar, es necesario realizar un diseño que marque las líneas generales del código. A fin de ilustrar y facilitar la comprensión de la solución software propuesta, se elabora un flujograma.

Dentro del software general del sistema, se pueden diferenciar dos códigos diferenciados: el programa de maestro y el programa de esclavo.

El maestro de la red es un módulo WiFi que conoce todas las instrucciones y rutinas de efectos que tanto las vidrieras como el rosetón deben ejecutar. Además de la información relativa a los colores, también conoce la sincronización temporal de los mismos. Centralizando la coordinación de todos los paneles en un solo módulo se obtiene una sincronía perfecta para los ojos de los espectadores. El maestro hace las veces de director de la orquesta que forman los 37 nodos esclavos adheridos a la red. De este modo, es el encargado de preparar los datagramas UDP. Esta preparación se lleva a cabo a través de algoritmos de ordenación y asignación de los caracteres del mensaje.

Por su parte, los módulos esclavos deben ser capaces de recibir el mensaje que el maestro ha enviado y filtrarlo para recoger la información que les atañe. Una vez han procesado esta información, su último objetivo es controlar la alimentación de las tiras LED a través de los pines destinados a ello. Por tanto, se podría decir que los módulos esclavos solamente saben "*cómo hacer*", pero no "*qué hacer*".

El filtrado del mensaje recibido en los esclavos se realiza indicando en la cabecera del código de cada módulo qué elementos de la cadena de caracteres deben recogerse. Esto es necesario ya que como ya se ha comentado previamente, la estrategia de comunicación será enviar mensajes en broadcast.

De acuerdo a la estructura predefinida de los programas para Arduino, el código se divide en dos bloques: setup y loop.

En el bloque setup se realizan los protocolos de conexión a la red local creada por el Punto de Acceso configurado previamente. En el caso de los módulos esclavos, se añade un método indicador del estado de dicha conexión. Mientras no logra conectarse a la red, cada 10 segundos mantiene las tiras LED encendidas en naranja durante 400 milisegundos. Una vez se haya conectado, se encienden 1 segundo en verde y se apagan a la espera de recibir mensajes del módulo maestro.

El bloque loop se llama así porque es un bucle. Se realiza infinitas veces una vez se ha ejecutado el setup del programa. En este bloque, el módulo maestro debe leer la siguiente instrucción y preparar el mensaje que enviar a los módulos esclavos, que deben recibir esa información y controlar las tiras LED en consecuencia.

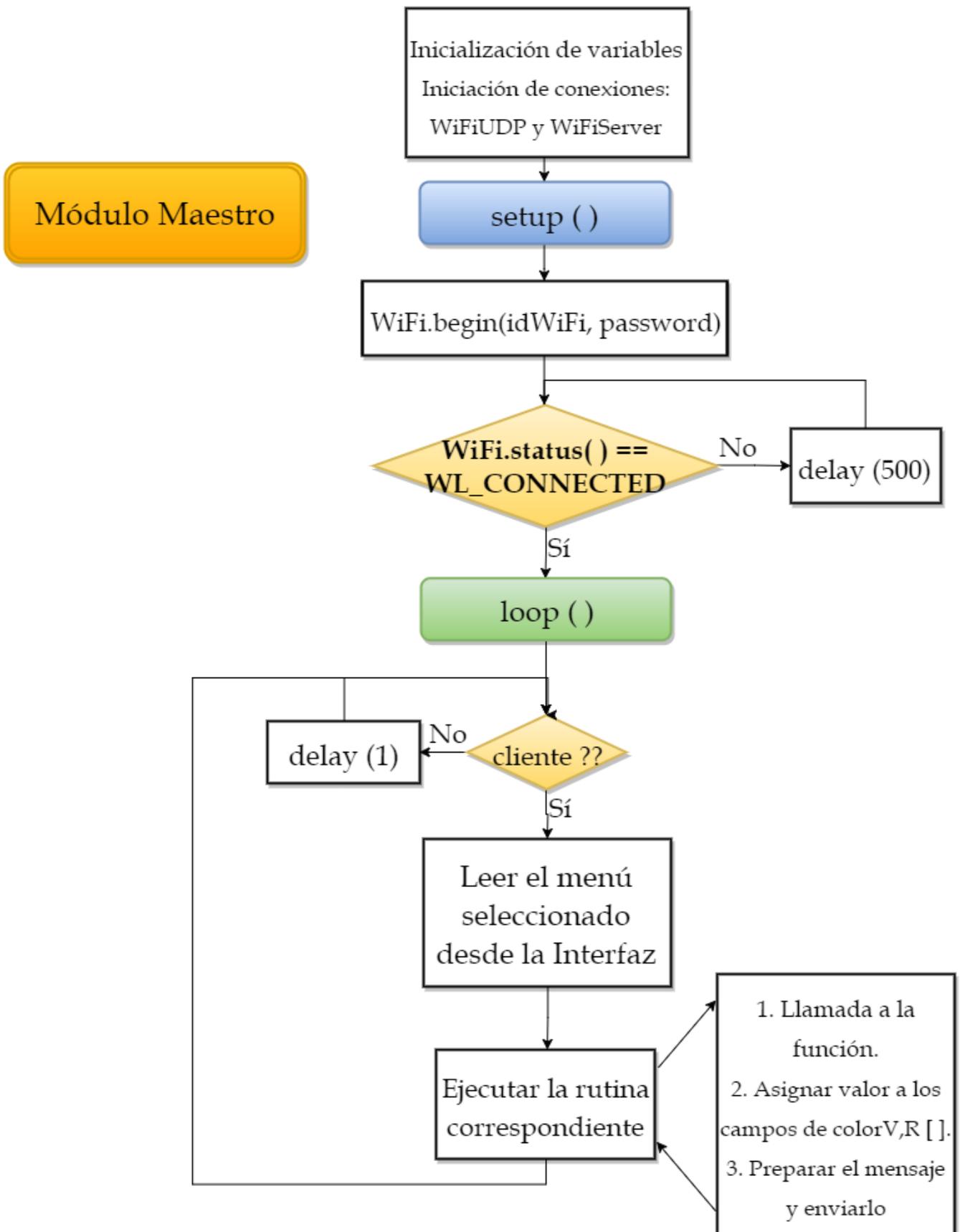


Figura 32: Diseño conceptual del programa del Módulo Maestro

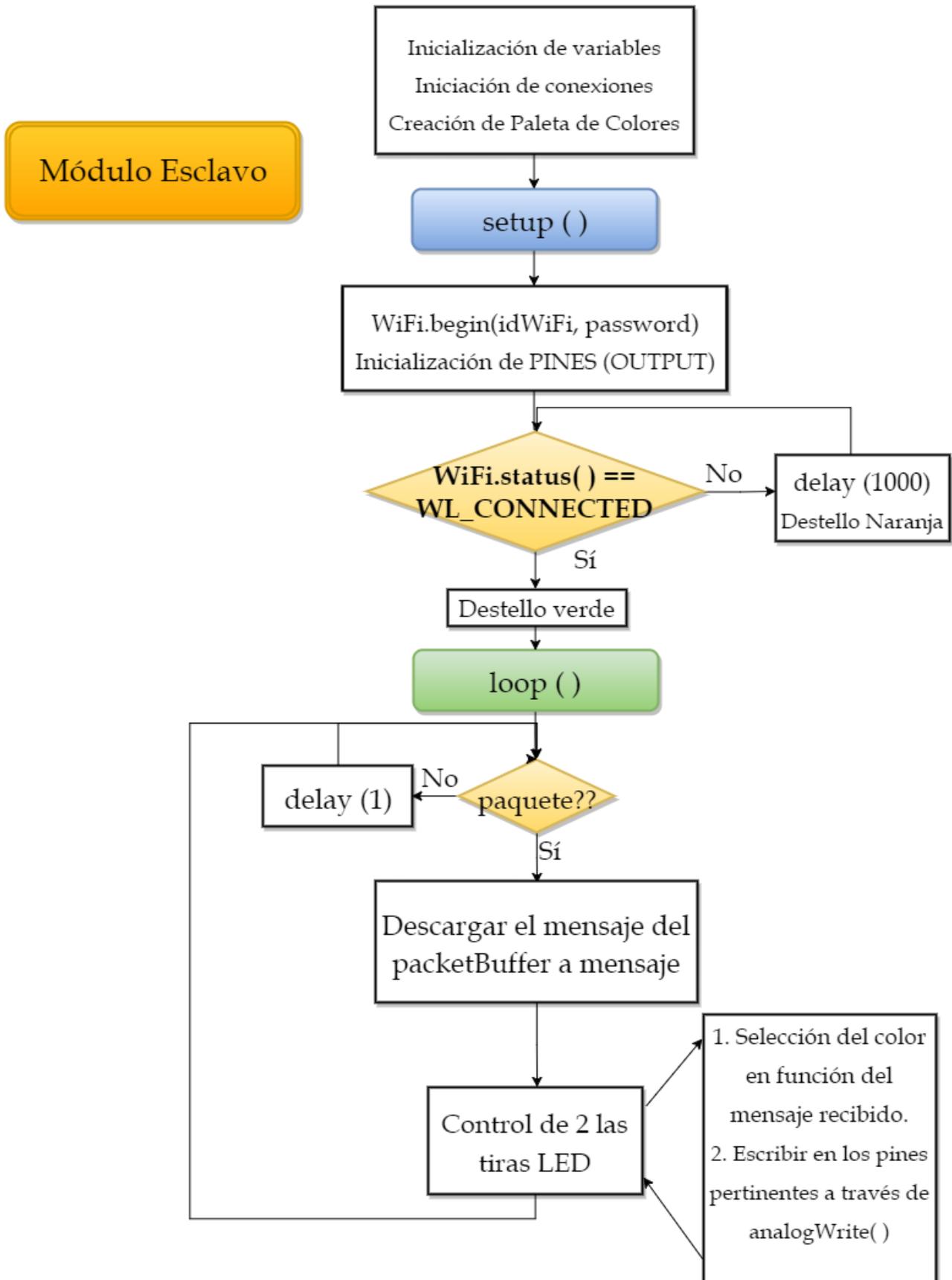


Figura 33: Diseño conceptual del programa de un Módulo Esclavo

## 6.4 Implementación de la Propuesta

La implementación del diseño para el software de control de la iluminación de las vidrieras y el rosetón, como se ha visto en capítulos anteriores, es diferente según se trate del módulo maestro, o de los módulos esclavos.

Sin embargo, hay puntos en común entre ambos. Independientemente del tipo de nodo, ambos deben declarar las variables que van a utilizar para sus funciones y rutinas. Esta declaración de variables se debe realizar en la cabecera del código, después de la inclusión de las librerías.

Entre estas variables comunes, se encuentran los datos de acceso a la red de área local y la más importante, la cadena de caracteres *mensaje[28]*. Esta cadena de caracteres tiene su origen en el módulo maestro. Él es el encargado de rellenar los elementos de dicha cadena en función de la combinación de colores que se haya programado. Una vez está completa, se envía a todos los módulos conectados a la red de área local. Sin embargo, esta cadena no se envía en forma de cadena de caracteres como tal, sino que la librería *WiFiUdp* la convierte en un datagrama que sí puede ser transmitido por radiofrecuencia.

Además de este puente de unión, otro punto en común entre ambos códigos es el bloque de conexión a la red WiFi e inicialización de los puertos para la comunicación a través del protocolo UDP. Este proceso se realiza una sola vez después de iniciar el módulo en el bloque de setup. Las sentencias correspondientes a esta conexión son las siguientes:

```
WiFi.begin(idWiFi,password);
while (WiFi.status( ) != WL_CONNECTED) {
    Serial.print("Attempting to connect to ");
    Serial.println(idWiFi);
    delay (500); // 10000 para los esclavos }
Serial.println ("Iniciando UDP");
Udp.begin ( localPort );
```

Cuando se ha conectado el maestro, imprime también en el Monitor Serial la dirección IP que se le ha asignado. Esto es necesario para conocer la dirección IP del servidor Web que soportará la Interfaz Gráfica de Control.

En ambos programas, tras la conexión a la red se llama a una función de nombre *printWifiStatus( )* que muestra en el Monitor Serial el estado de la conexión incluyendo el nombre y dirección IP del Punto de Acceso así como la Potencia transmitida (RSSI) en dBm.

## 6.4.1 Programa Maestro

El programa del módulo maestro presenta tres bloques principales. En primer lugar, da soporte al Servidor Web donde se encuentra la Interfaz Gráfica de Control. A partir de las solicitudes de los clientes de la Web, debe activar una rutina de iluminación u otra. Por último, cuenta con la memoria de programa del conjunto. De una manera similar al incremento del *Program Counter* en la CPU para conocer cuál es la siguiente instrucción que debe realizar un microprocesador, el módulo maestro debe ir recorriendo secuencialmente una memoria de programa que va indicando los diferentes cambios de iluminación.

En este apartado se abordan el segundo y tercer bloque, dejando para capítulos sucesivos lo referente al servicio de la Interfaz Gráfica de Control.

La lectura de la solicitud del cliente de la Interfaz es sencilla y se basa en el reconocimiento de la última parte de la URL asociada a la rutina que corresponde. Esto se realiza en dos pasos:

- i. Leer y almacenar en una cadena de caracteres la dirección a la que el cliente solicita acceso.
- ii. Rastreo del número del programa dentro de esta cadena a través de una función que devuelve un '-1' cuando no se ha encontrado.

Para almacenar la solicitud se emplea un método de la Clase **Client**. Esta clase pertenece a la librería *ESP8266WiFi*. El método devuelve una cadena de caracteres cuyo contenido viene dado por la primera línea de la respuesta del cliente. Después se llama a otro método que espera hasta que todos los caracteres de salida del buffer se han enviado correctamente:

```
String req = client.readStringUntil('\r');
client.flush( );
```

Entonces se activan unas sentencias condicionales para identificar cuál ha sido el programa seleccionado a través de la función `indexOf( )`. Concretamente, `indexOf( )` devuelve el índice de su posición en la cadena de caracteres de su último elemento. Si no se identifica dicho carácter dentro del string, devuelve un '-1'. Estas sentencias condicionales dan valor a una variable bandera que se comparará en un bloque *switch*.

Un bloque *switch* en C tiene como argumento una variable, que suele ser un carácter. Este valor se compara con diferentes opciones o *cases* y se ejecutan las sentencias correspondientes a aquella o aquellas que sean idénticas a la variable. Cuando tan solo se quiere ejecutar una de ellas, tras cada opción se sitúa un `break`, que conlleva la salida del bloque.

La estructura de las sentencias correspondientes es idéntica en todos los casos: la primera llama a la rutina de iluminación que el cliente quería activar y la segunda actualiza el valor de la variable a 0, para limpiar su contenido.

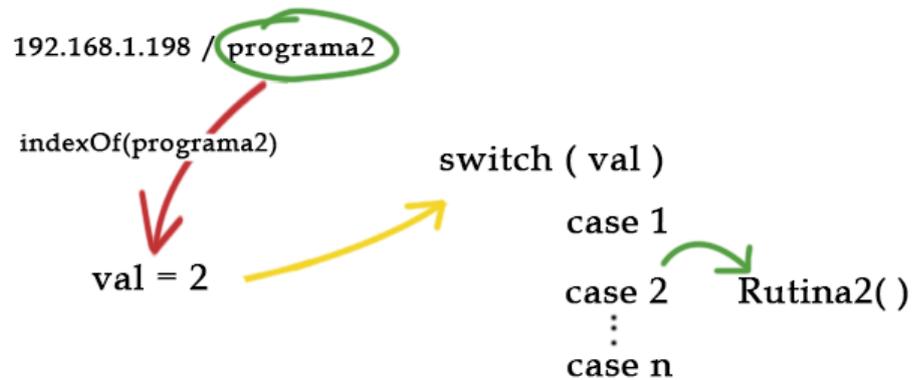


Figura 34: Lectura de la solicitud del cliente. Selección de la rutina

Una vez afrontada la selección de la rutina de iluminación, se procede a explicar cómo se generan estas rutinas y los algoritmos que preparan las combinaciones de colores.

En primer lugar, cada rutina de iluminación se corresponde con un número musical del espectáculo. Está formada por diferentes efectos y combinaciones de colores con sus temporizaciones. Estos efectos se suceden los unos a los otros conforme a la música y lo que ocurre en el escenario. De esta manera se consigue crear una escenografía dinámica que envuelve al espectador.

Para conseguir este propósito, hay que realizar un diseño de las diferentes combinaciones y efectos de iluminación para las tiras LED teniendo en cuenta sus posiciones. La preparación de los mensajes que las gobiernan se lleva a cabo de un modo más sencillo empleando una notación de índices para nombrar a todos los paneles. Estos índices corresponden con las posiciones de la cadena de caracteres que contiene la información de los colores. Así, se tienen dos cadenas: `colorV[16]` y `colorR[10]`. Concatenando las dos y un par de caracteres más se consigue construir el mensaje completo que se enviará en modo broadcast. Este mensaje además contiene dos caracteres iniciales que se reservan para enviar información extra a los módulos, por ejemplo, para diferenciar el efecto que realizan las vidrieras delanteras de las traseras.

Los algoritmos de asignación de caracteres utilizan en su mayoría bucles de tipo *for* para rellenar los elementos de la cadena y bucles de tipo *while* para llevar el control de la temporización. Estas subrutinas o efectos son funciones que actualizan los valores de las cadenas `colorV[ ]` y `colorR[ ]`. Típicamente estas funciones cuentan con al menos tres parámetros: un carácter de selección del color, un entero para la temporización y la cadena que se pretende modificar.

Sin embargo, algunas funciones necesitan otros parámetros. Por ejemplo, otro carácter que selecciona un color con el que hacer un barrido sobre un color base.

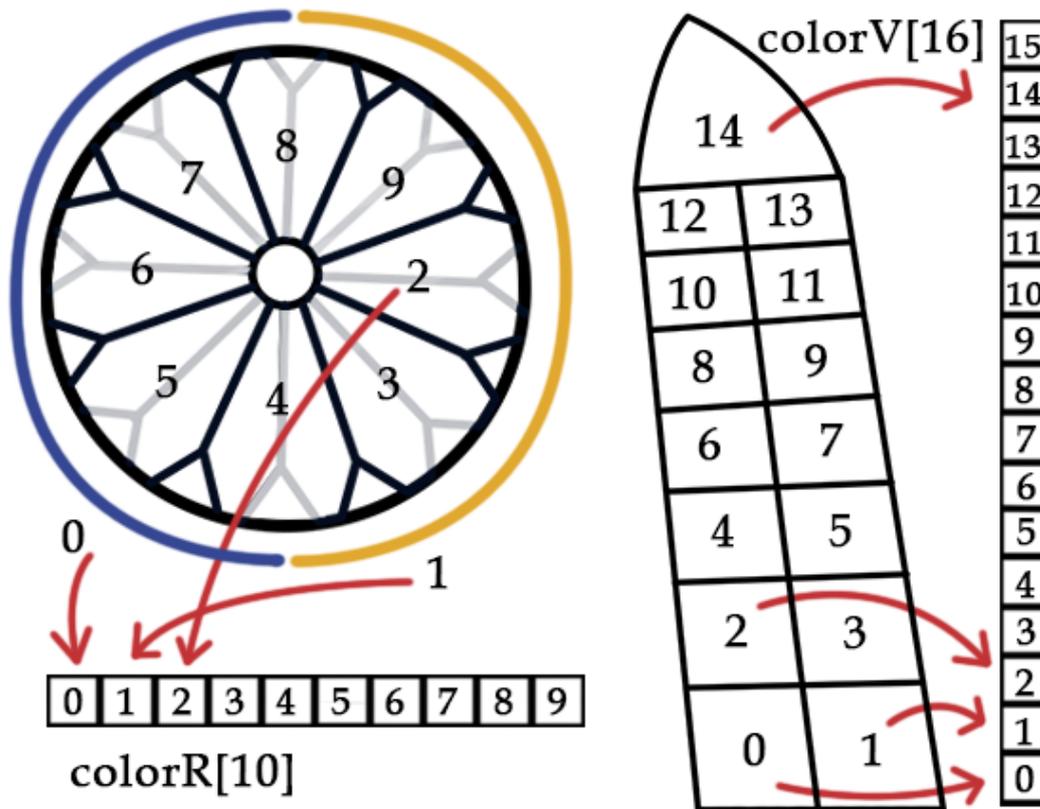


Figura 35: Notación de índices para la programación de colores en las estructuras.

El empleo de esta notación de índices contribuye a una programación más clara y precisa. Para ilustrar el funcionamiento de la misma, se propone el siguiente caso:

Se pretende que todas las vidrieras se iluminen de color verde durante 5 segundos y el rosetón se mantenga en azul. Después se quiere que se apaguen completamente hasta recibir una nueva orden.

Para conseguirlo, en primer lugar hay que dar valores a los elementos de las cadenas de caracteres `colorR[ ]` y `colorV[ ]`. Una vez estén asignados todos los campos, se copian ambas cadenas en `mensaje[ ]` para enviarlo a todos los módulos. El contenido de la subrutina sería así:

```
for( i = 0; i < 16; i++) {
    colorV[i] = 'G'; }
for( i = 0; i < 10; i++) {
    colorR[i] = 'B'; }
```

Los caracteres 'G' y 'B' hacen referencia al color verde y azul respectivamente (green y blue en inglés). Ya se ha cargado la información relativa a los colores de las tiras. Entonces es necesario preparar el mensaje que se va a enviar. Para ello, se introducen los valores de `colorV[ ]` y `colorR[ ]` en la cadena `mensaje[ ]` reservando previamente dos elementos para instrucciones adicionales. Por tanto, esta subrutina es fija sea cual sea la combinación de colores que se quiera imponer. Estará contenida dentro de la función `sendSetColor()`:

## 6. Desarrollo Software

```
void sendSetColor( char IP[ ], int localPort, char *colorR,
                  char *colorV, int time) {

//se prepara el mensaje
    for ( i =0; i <16; i++ ) {
        mensaje[i+2] = colorV[ i ]; }
    for (i=0; i < 10; i++) {
        mensaje[i+18] = colorR [i]; }

//se envía el paquete
    Udp.beginPacket( IP, localPort);
    Udp.write(mensaje);
    Udp.endPacket( );

        delay ( time ); }
```

La función sendSetColor es la encargada de enviar todos los paquetes una vez se han preparado. Esta preparación es llevada a cabo por funciones que producen los efectos visuales que se persiguen. Por ejemplo, hacer barridos con los colores o iluminar las vidrieras como si fuesen un tablero de ajedrez. Algunas de estas funciones son:

<b>Nombre</b>	<b>Efecto</b>
Full (char c, int time, char *color)	Ilumina todos los elementos en el color que indica el parámetro char c
BlinkingCols (char a, char b, int time, char *colorV)	Exclusiva para vidrieras. Enciende cada columna de un color y las alterna según el valor de time.
Chess (char a, char b, int time, char *colorV)	Exclusiva para vidrieras. Ilumina las vidrieras como un tablero de ajedrez en los colores indicados.
Fan (char a, char b, int time, char *colorR)	Exclusiva para el rosetón. Efecto de ventilador. Un aspa va girando en el color 'a' sobre una base de color 'b'.
FixedCross ( char a, char b, int time, char *colorR)	Forma en el rosetón una gran cruz iluminando los sectores 2, 4, 6 y 8.
Striped (char a, char b, int time, char *colorR)	Enciende las vidrieras con rayas horizontales alternas en los colores indicados.

## 6.4.2 Programa Esclavo

Las labores principales de los módulos esclavos son dos: recibir paquetes a través de su conexión WiFi y gobernar los colores de las tiras LED RGB. Por tanto, vamos a estudiar los bloques del código que realizan esta dos funciones.

Se recuerda que un módulo es capaz de manejar dos tiras LED ya que cada una necesita tres pines configurables con PWM para su alimentación. Por tanto, hay que inicializar los seis pines como salida.

```
pinMode(rled1, OUTPUT);
pinMode(gled1, OUTPUT);
pinMode(bled1, OUTPUT);
pinMode(rled2, OUTPUT);
pinMode(gled2, OUTPUT);
pinMode(bled2, OUTPUT);
```

Cuando se han configurado como salida, ya están listos para utilizar el PWM. Para ello, se utiliza la función `analogWrite(int pin, int value)`, estando el valor comprendido entre 0 y 255.

Una vez se ha conectado a la red WiFi, los módulos esclavos deben estar esperando la llegada de un paquete que contenga la información que necesitan para iluminar las tiras. El módulo detecta la llegada de dicho mensaje cuando al comparar el tamaño del paquete, éste es distinto de 0.

```
int packetSize = Udp.parsePacket( );
if (packetSize) { ... }
```

Cuando ha llegado un paquete, este debe ser descargado en una cadena de caracteres que también se conoce como `mensaje[ ]`. Se recuerda que dicho mensaje porta la información relativa a todas las tiras LED del conjunto. Por tanto, cada módulo debe seleccionar el color de los paneles que controla atendiendo a los elementos de la cadena de caracteres que correspondan. Para facilitar la programación de la treintena de módulos esclavos, estos parámetros se introducen a través de una definición en la cabecera del código. El valor que se debe dar a las constantes `C_0`, `C_1` es igual a la posición de los caracteres que dan color a sus dos tiras. Por ejemplo, para las tiras 0 y 1 de la vidriera:

```
#define C_0 2
#define C_1 3
```

Conocidos los caracteres que indican el color de las tiras, hay que traducir esta información para obtener los valores del PWM de cada pin. Se ha programado una función que cumple esta misión. La función cuenta con dos parámetros y su prototipo es:

```
void selectColor (int color[3], char c) { ... }
```

## 6. Desarrollo Software

El vector de enteros `color[ ]` se debe rellenar según cuál sea el carácter 'c'. Esta selección se hace a través de un bloque switch que enfrenta los caracteres recibidos con la paleta de colores declarada en la cabecera del programa. Esta gama de colores almacena los distintos colores que se desean utilizar en esta aplicación y sus valores para el PWM:

Carácter del Mensaje	int color[3]={Red, Green, Blue}
	R      red[ ]= { 255, 0 , 0 }
	G      green[ ]= { 0, 255, 0 }
	B      blue[ ]= { 0, 0 , 255 }
	Y      yellow[ ]= { 255, 255 , 0 }
	P      pink[ ]= { 255, 51 , 255 }
	O      orange[ ]= { 255, 120 , 0 }
	V      violet[ ]= { 120, 0 , 120 }
	T      turquoise[ ]= { 0, 0 , 255 }
	W      white[ ]= { 255, 255 , 255 }
	N      off[ ]= { 0, 0 , 0 }

Cuando se conocen los valores del PWM de cada uno de los pines, se llama a otra función que es capaz de escribirlos para alimentar los LED. Estos valores se mantienen constantes hasta que el módulo esclavo recibe otro mensaje. Entonces, se repite el proceso que se ha descrito en este subapartado.

Capítulo 7

# **INTERFAZ GRÁFICA DE CONTROL**

Diseño, programación y funcionamiento

## 7.1 Diseño y Programación de la Interfaz

Cualquier aplicación necesita una zona de comunicación entre el usuario y el propio sistema. Aunque no es un aspecto fundamental de este proyecto, mientras se trabajaba en el diseño del sistema se vio la necesidad de crear un espacio de interacción que permitiese activar las diferentes rutinas. De este modo, el técnico del espectáculo encargado de la iluminación en el escenario podrá hacer coincidir el inicio de estas rutinas con la música del espectáculo. Este apartado aborda la solución final adoptada para resolver esta cuestión.

En primer lugar se evaluó la posibilidad de construir una interfaz física. Este dispositivo podría estar formado por una botonera que, actuando como periférico del módulo maestro, indicaría cuál es la secuencia de rutinas que se debería reproducir. También se barajó la posibilidad de añadir un visualizador formado por displays de 7 segmentos o incluso una pantalla LCD. La ventaja de estas interfaces es que al estar conectadas a través de conexiones físicas al módulo, no se generan retrasos asociados a la comunicación.

Sin embargo, la opción de crear una interfaz física se desechó fundamentalmente debido a dos razones. La primera de ellas se debe a la complejidad adicional que resulta del diseño de un componente hardware de estas características. Aunque su construcción habría sido realmente interesante, se escapa de los objetivos de este proyecto. La segunda razón son motivos económicos: una interfaz física consume muchos más recursos que una interfaz software. Entre estos recursos se encuentra el tiempo y el dinero.

Dicho esto, la mejor solución parece ser una interfaz software que permita comunicar al usuario con el módulo maestro a través de un ordenador o una tableta táctil.

Esta interfaz debe ser compatible con los microprocesadores del proyecto. Aprovechando la red local ya creada para la comunicación entre ellos, se determina el uso del WiFi para dar órdenes al módulo maestro. Por tanto, de entre todas las posibilidades, se adopta la solución de crear una aplicación Web.

Para comenzar el diseño de este sitio Web se establecen las funcionalidades básicas que dicha interfaz debe satisfacer:

- i. Mostrar un menú de selección de programas para elegir una rutina según el número musical que corresponda durante el espectáculo.
- ii. Ofrecer también otras rutinas adicionales. Por ejemplo, una rutina de comprobación previa al comienzo de la función.
- iii. Cumplir las dos funcionalidades respetando algunos aspectos estéticos que hagan que la interfaz sea sencilla e intuitiva.

Cuando ya se tienen claros los aspectos que debe resolver la interfaz, hay que preparar las herramientas de programación. Entre estas herramientas se encuentran los lenguajes de programación HTML y CSS.

[28]HTML (*HyperText Markup Language*) es un lenguaje de etiquetas que se utiliza para programar páginas web. Las etiquetas de HTML son palabras clave comprendidas entre paréntesis angulares ( < y >). Habitualmente van por parejas y la segunda suele aparecer con una barra slash ( />). HTML es un lenguaje muy sencillo que permite describir documentos hipertexto. Esto quiere decir que se trata de documentos cuyo texto presenta diferentes aspectos estéticos como la posición, el color o el tamaño.

Aunque HTML permite dar formato a estos textos, el estilo se aplica de un modo mucho más efectivo utilizando CSS (*Cascading Style Sheets*). [29] CSS es un lenguaje de hojas de estilo que describe la presentación de los documentos HTML. Explica como los elementos deben ser mostrados en la pantalla del navegador, en papel, o en otros medios.

Para comenzar a trabajar, además de algunas nociones básicas sobre estos dos lenguajes de programación, se necesita un editor de texto y un visualizador o navegador Web. El editor de texto se utiliza para escribir los documentos HTML, que serán posteriormente interpretados por el navegador. Durante el desarrollo de esta interfaz, se ha utilizado el software libre Notepad ++ para la edición del texto. Así como el editor de texto usado no afecta a la apariencia final del sitio, no todos los navegadores Web presentan los documentos exactamente del mismo modo. Fundamentalmente se debe a factores como la resolución de la pantalla o pequeñas variaciones del color aunque, en ocasiones, no son capaces de leer algunas de las etiquetas.

Algunas de las etiquetas básicas empleadas para describir la interfaz gráfica de la aplicación son las siguientes:

<b>&lt;Etiqueta&gt;</b>	<b>Función</b>
<html> </html>	Inicio y cierre del documento HTML
<head> </head>	Encierra la información del documento que no se ve en la pantalla principal. Por ejemplo, el título del documento o los estilos.
<body> </body>	Dentro del cuerpo del documento HTML se incluye cualquier carácter imprimible.
<p> </p>	Separador de párrafos en HTML.
<div> </div>	Encierra secciones dentro del cuerpo del documento.
<button> </button>	Convierte su contenido en un botón típico de los navegadores. Se le ha dado estilo para cuidar la estética.
<a> </a>	Utilizado para redirigir al usuario a la página de cada rutina, al enlace que lee el servidor.

## 7.2 Funcionamiento de la Interfaz

Los objetivos que se perseguían desarrollando la interfaz software para el control de la aplicación quedan satisfechos tras su puesta en marcha. Efectivamente se ha logrado activar las diferentes rutinas desde un ordenador conectado a la red de área local creada para las comunicaciones.

Para ello, la página es soportada por el servidor Web que ha establecido el módulo maestro. Una vez se ha programado y probado desde el ordenador, es necesario introducir en el código del Programa Maestro las instrucciones pertinentes para la respuesta del servidor a los clientes.

Esta respuesta se prepara introduciendo en una cadena de caracteres el contenido del documento HTML que ha sido codificado previamente. Cuando se encuentra almacenado en dicha cadena, se llama a una función de la clase **Client**. El nombre de esta función es **client.print** y su único parámetro es la cadena de caracteres que se pretende imprimir en la pantalla del navegador del cliente. Después de haberse cargado completamente el documento HTML, el cliente se desconecta del servidor y el objeto 'cliente' se destruye.

Como se ha explicado, el módulo maestro actúa como servidor del sitio Web y por tanto, el acceso al mismo se realiza a través de una URL que contiene su dirección IP. Además, como este servidor forma parte de una red de área local, el cliente y el maestro deben estar conectados a la misma red.



Figura 36: Dirección Web de la Interfaz Gráfica de Control

El acceso a este sitio presenta dos controles de acceso para mejorar sus condiciones de seguridad. El primero de ellos, como se ha comentado en el párrafo anterior, es la necesidad de pertenecer a la misma red. Esta red ya cuenta con una contraseña de acceso que limita la entrada de usuarios indeseados. Además, se prepara un acceso privado a través de otro documento HTML que solicite datos de usuario y clave para entrar en el sistema.

La estructura de la interfaz se divide en cuatro bloques principales:

- i. **Cabecera.** Está formada por una franja negra que contiene el título del sitio: "Interfaz Gráfica de Control v2.0" y el sello institucional del usuario de este sistema.
- ii. **Bloque de Rutinas.** Se sitúa en el sector izquierdo de la pantalla y muestra las diferentes rutinas para los números musicales del espectáculo. Los nombres que aparecen hacen referencia a los títulos de las canciones a las que corresponden.

- iii. **Bloque de Rutinas Adicionales.** Se encuentra dispuesto en el lado derecho de la pantalla. Incluye el rótulo del espectáculo y un cuadrante con 6 botones de selección. El primero de ellos activa una rutina de pruebas que certifica el correcto funcionamiento de todo el sistema antes del comienzo del espectáculo. Inmediatamente debajo se encuentran 5 botones que iluminan todos los elementos del sistema con un color fijo. El último botón se emplea para apagar todas las tiras LED.
- iv. **Pie.** Franja estrecha que enmarca el conjunto e indica la autoría del desarrollo de la Interfaz.

Todas las rutinas se activan pulsando en el botón correspondiente. Estos botones son realmente enlaces que no redirigen a otro sitio diferente. Sino que envían al microprocesador una solicitud en formato dirección URL de la que se extrae el último segmento. Este último tramo de la solicitud, como se explica en el capítulo de Desarrollo Software, indica la rutina que el cliente del sitio Web ha seleccionado.



Figura 37: Interfaz Gráfica de Control

## Capítulo 8

# **RESULTADOS, CONCLUSIONES Y LÍNEAS FUTURAS**

## 8.1 Resultados y Conclusiones

Tras la realización física y desarrollo del proyecto escenográfico que se describe en la presente Memoria, se puede concluir que la solución adoptada satisface las necesidades exigidas por el espectáculo.

En líneas generales, las herramientas seleccionadas para la programación del software y el montaje físico han resultado adecuadas. Sin embargo, en algunos aspectos sí han presentado carencias que han limitado la robustez y prestaciones del sistema. Entre ellas, el tiempo de respuesta del sistema.

Además de estas deficiencias técnicas, que se desarrollarán posteriormente, cabe destacar otras problemáticas surgidas a raíz del trato con proveedores y otros profesionales. Como gerente del proyecto, se ha ratificado la extrema importancia de un adecuado control de tiempos y recursos. Un control detallado y exhaustivo del avance del proyecto garantiza la conclusión de las actividades dentro del plazo predefinido. Además, en general contribuye a reducir los sobrecostes derivados de una mala gestión de los recursos.

No obstante, estas circunstancias no han sido la tónica general del proyecto, que ha marchado a un ritmo mayoritariamente homogéneo.

Las limitaciones técnicas asociadas a la selección de herramientas y tecnologías más fundamentales son los retardos observados durante las pruebas en el emplazamiento final de la instalación. Aunque no se pueda considerar que comprometan la fluidez del avance de las rutinas, es innegable el ligero desfase temporal que se produce al activar cualquiera de ellas desde la Interfaz Gráfica de Control. Estas condiciones deben tenerse en cuenta para el desarrollo final de las coreografías de iluminación que serán expuestas durante el espectáculo.

Realizando un profundo estudio con objeto de encontrar el cuello de botella en el proceso de transmisión de la información, se concluye que, efectivamente, hay algunos puntos del sistema más propensos a generar estos retrasos.

En primer lugar se desprecian los retrasos generados en el hardware del sistema. Es decir, no se considera que el tiempo de encendido de los LED pueda dar pie a un retraso perceptible a través del sistema de visión humano. Tampoco se cree que los desfases temporales se estén produciendo en el circuito de excitación de los LED ya que los transistores empleados son de alta velocidad de respuesta.

Tras estas consideraciones, todo apunta a una pérdida de simultaneidad en las comunicaciones inalámbricas. Estos desfases, por tanto, se pueden deber a varios factores. El primero de estos factores es la intensidad de la señal de radiofrecuencia que fundamenta la comunicación inalámbrica.

Las condiciones que afectan a la potencia de esta red son, por ejemplo, obstáculos físicos como muros y personas. A fin de resolver esta problemática,

## 8. Resultados, Conclusiones y Líneas Futuras

se trata de situar el router en un lugar alto, tan equidistante como sea posible a los módulos esclavos y de tal modo que se encuentre el menor número de impedimentos físicos. Además de estas barreras materiales, la potencia de la propia red depende de la capacidad del Punto de Acceso y también de las antenas receptoras de los módulos esclavos.

Para concluir con la cuestión de los retrasos acumulados en la comunicación, se señala a la Interfaz Gráfica de Control como principal responsable: el tiempo de respuesta del servidor web no puede ser despreciado en algunas ocasiones. Como estrategia para minimizar estos retrasos, se procederá a la conexión vía Ethernet del dispositivo con el Punto de Acceso. De esta manera, se pretende mejorar la conectividad del ordenador dentro de la red.

Dejando al margen las pequeñas demoras ante las instrucciones del usuario, que en ocasiones son imperceptibles, el resultado global del sistema es más que satisfactorio.

La iluminación obtenida presenta unos colores muy atractivos que consiguen crear diferentes ambientes y sensaciones en el público. Gracias a la labor técnica desarrollada, se potencia la labor artística del elenco de la producción.

El conjunto genera, por tanto, efectos de iluminación que impresionarán sin duda a todos los espectadores, dando un acabado realmente profesional a la producción. De hecho, es indiscutible el parecido entre las soluciones adoptadas en la mayor parte de las producciones profesionales y los resultados del proyecto desarrollado.

Teniendo en cuenta que se trata de una producción musical completamente amateur y sin ánimo de lucro, el producto final obtenido a través del trabajo desarrollado en este proyecto supera las expectativas iniciales.

## 8.2 Líneas Futuras

Este apartado pretende abordar las posibilidades que se abren una vez se ha alcanzado el término del presente proyecto. Se considera que señalar posibles líneas de desarrollo es uno de los puntos con más interés de toda la Memoria.

Cuando ya se ha puesto en marcha un producto completo que satisface los objetivos iniciales, aparecen una gran cantidad de líneas de desarrollo del producto. Todas ellas conducen a mejorar diferentes aspectos del mismo concediendo al sistema un grado extra de fiabilidad, robustez o profesionalidad. Después de haber invertido tantos esfuerzos y recursos, se considera realmente atractivo continuar con las mejoras que el producto podría necesitar.

Habiendo evaluado previamente los resultados obtenidos, y sus fortalezas y debilidades, se encuentran varias líneas de trabajo futuro para este sistema.

- i. Desarrollo de un software más potente que el logrado hasta la fecha.**  
Este software, idealmente desarrollado para ordenador, podría ofrecer, entre otras prestaciones, un modo de edición de rutinas más intuitivo que el método actual. Como si se tratase de la edición de un video, sería muy interesante una interfaz que trabajase con bloques y líneas temporales. Esto permitiría a cualquier usuario, incluso no iniciado en programación, la creación de rutinas para el sistema.
- ii. Mejoras en la Interfaz Gráfica de Control.**  
Se propone un control más pleno del avance de las rutinas por parte de los técnicos de iluminación. De este modo, el usuario podría ser capaz de pausar y posteriormente reanudar el progreso de los efectos. Incluso seleccionar el punto de comienzo de las mismas.
- iii. Aumento del control de los nodos de la red.**  
A través de un manejo más exhaustivo de los direccionamientos IP, desarrollar un método que garantizase que todos los módulos esclavos tienen una conexión a la red de buena calidad. Además, también se podría conocer qué módulos se encuentran defectuosos en caso de producirse un fallo en el sistema.
- iv. Empleo de una comunicación que contemple el acuse de recibo.**  
Las causas de algunos posibles fallos en el recibo de mensajes podrían conocerse si el sistema informase de comportamientos inesperados durante el envío. De este modo, si el técnico recibiese un 'OK' con la llegada de todos los mensajes, podrían detectarse las circunstancias que producen estas errores en la comunicación.

Capítulo 9

# **GESTIÓN DEL PROYECTO**

Planificación Temporal y Presupuesto

## 9.1.1 Diagrama EDP

La estructura de descomposición de este proyecto se representa gráficamente en la siguiente figura. En ella se contemplan las actividades que ha conllevado la elaboración del presente TFG.

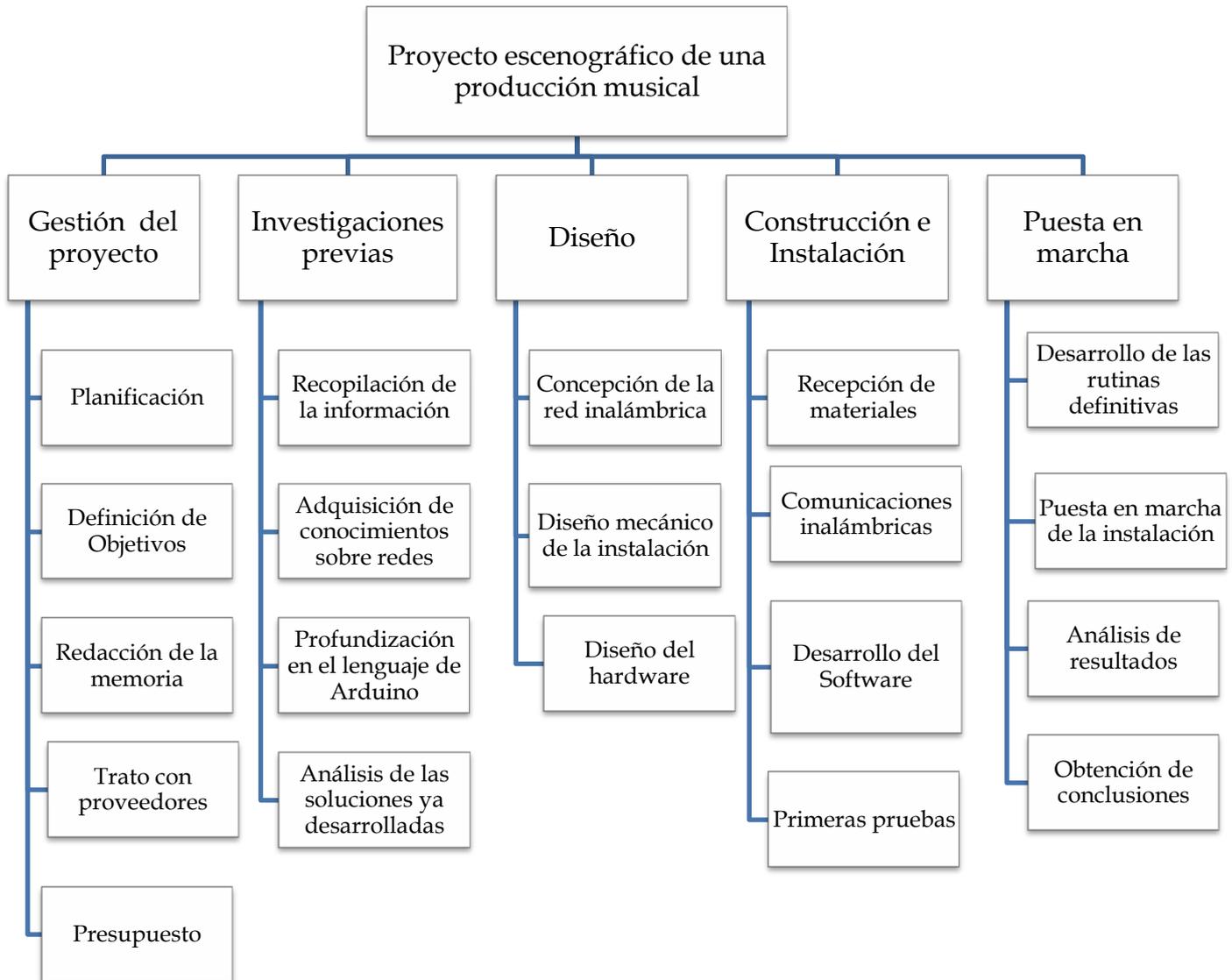


Figura 38: Diagrama EDP del proyecto.

## 9.1.2 Diagrama de Gantt

El diagrama de Gantt es una herramienta gráfica cuyo objetivo principal es exponer el tiempo de dedicación previsto para las diferentes actividades que conforman un proyecto. Estas actividades se señalan en el apartado anterior a través del diagrama EDP, que no contempla la temporización de estas actividades.

## 9. Gestión del Proyecto

En vistas al desarrollo definitivo del proyecto, la planificación temporal sufrió modificaciones debido a factores externos. La fecha del estreno se trasladó hasta después del verano, en octubre. Si es cierto que, al prolongar este plazo de entrega, todas las actividades se pudieron realizar con más detenimiento y, por tanto, se obtuvieron mejores resultados.

Conforme al mismo, las actividades realizadas pueden estructurarse en 5 bloques: gestión del proyecto, investigaciones previas, diseño, construcción e instalación y puesta en marcha.

Cabe destacar que las actividades se han realizado de manera homogénea a lo largo del desarrollo del proyecto, aunque los dos últimos meses de un modo más intenso. Esto se debe a dos razones fundamentalmente. En primer lugar, los retrasos ocasionados por los plazos de entrega de los proveedores (diseño gráfico, estructuras...). En segundo lugar, los desfases temporales asociados a no cumplir con excesivo rigor esta planificación. A fin de tener en cuenta la disponibilidad temporal real, se programaron descansos durante los periodos de exámenes de enero y junio.

Actividad	Start	Fin	Dur...
<b>Elaboración del Proyecto</b>	<b>25/10</b>	<b>31/07</b>	<b>281d</b>
<b>Gestión del Proyecto</b>	<b>02/11</b>	<b>22/07</b>	<b>211d</b>
Definición de Objetivos	02 /11	22/11	21d
Planificación Temporal	28/11	07/12	10d
Presupuesto	15/11	20/12	35d
Redacción de la memoria	02/02	22/07	143d
<b>Investigaciones previas</b>	<b>25/ 10</b>	<b>22/ 02</b>	<b>83d</b>
Recopilación de información	25/10	20/ 12	56d
Conocimientos sobre redes	28/11	22/2	30d
Lenguaje Arduino	19/ 11	28/11	8d
Análisis de soluciones ya de...	04/11	18/02	20d
<b>Diseño</b>	<b>01/03</b>	<b>22/04</b>	<b>53d</b>
Concepción de la red inalám...	04/11	19/05	16d
Diseño mecánico de la insta...	01/03	10/03	10d
Diseño del hardware	28/03	22/04	27d
<b>Construcción e Instalación</b>	<b>09/04</b>	<b>25/06</b>	<b>78s</b>
Comunicaciones inalámbricas	06/05	11/05	5d
Desarrollo Software	09/04	28/05	50d
Primeras pruebas	29/05	26/06	28d
<b>Puesta en Marcha</b>	<b>26/06</b>	<b>31/07</b>	<b>36d</b>
Desarrollo de las rutinas def...	26/06	31/07	36d
Análisis de resultados	26/06	15/07	19d
Obtención de conclusiones	18/07	22/07	4d

Figura 39: Tabla de actividades y su temporización

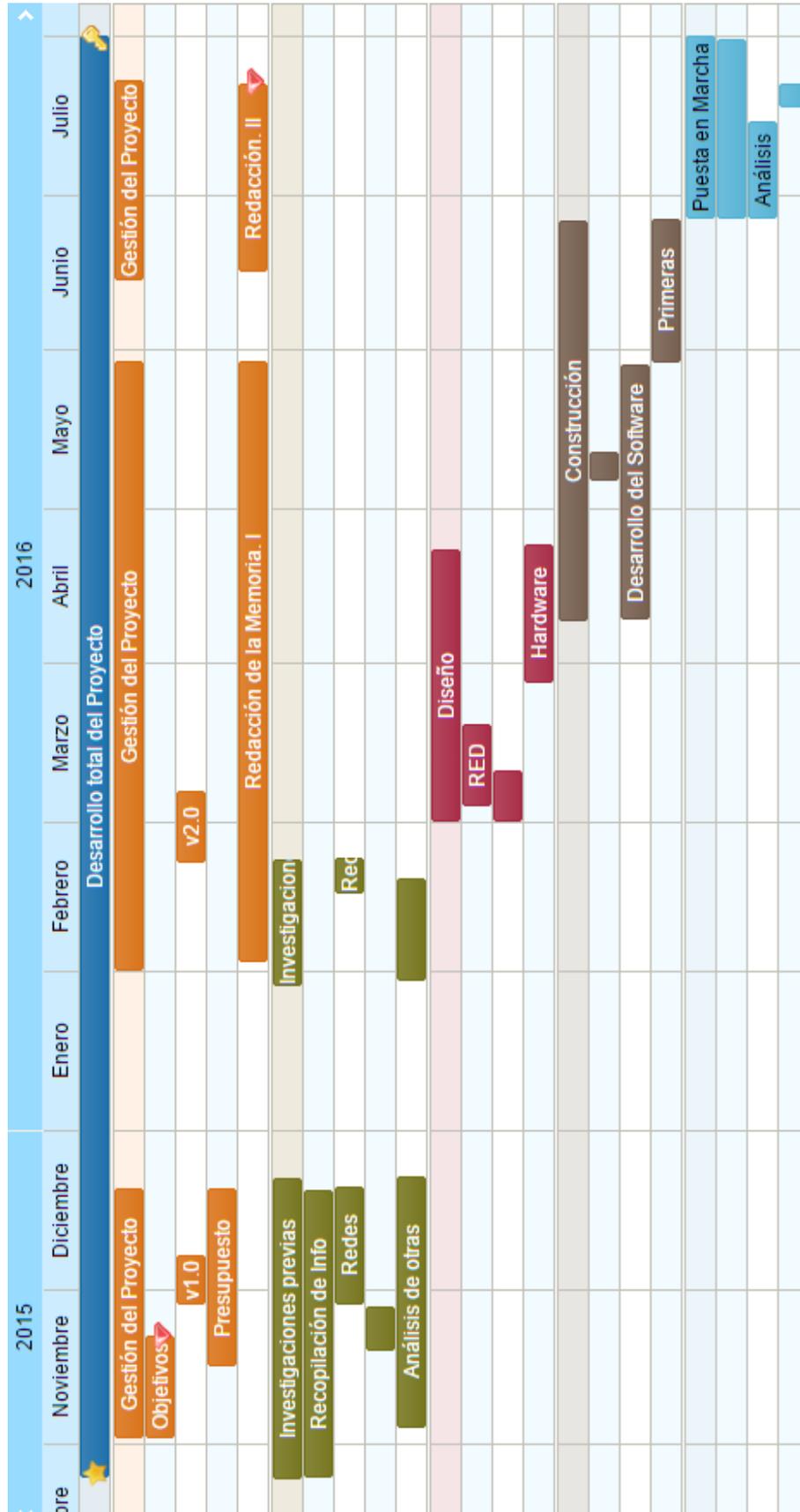


Figura 39: Diagrama de Gantt del proyecto

## 9.2 Presupuesto

Para continuar con la gestión del proyecto, es necesario tomar en consideración los recursos económicos que se consuman durante su desarrollo. En la siguiente ejecución del presupuesto se representan los gastos asociados a la puesta en marcha y construcción de la instalación. Se divide en dos bloques: costes asociados al material y equipo de la instalación eléctrica y costes debidos a la construcción e instalación de las estructuras. Los precios incluyen el IVA correspondiente:

### Componentes eléctricos y equipo.

Concepto	Precio Ud. (€)	Cantidad	Subtotal (€)
Fuentes de alimentación para PC	20.35	5	101.75
Tiras LED RGB (5m)	12.39	52	644.28
Kit para pruebas iniciales	90	1	90
Módulos WiFi	11	37	407
Drivers	10	72	720
Cable LC4C	0.45	100	45
Cable LC5C	1.05	20	21
Cable LC2C	0.3	20	6
Cable 0.5	0.25	50	12.5
Tubo termorretráctil 2.4	0.25	3	1.8
Tubo termorretráctil 3.2	0.6	2	2.24
Tubo termorretráctil 6.4	0.86	2	1.72
Pelacables	2.45	2	4.9
Pines	1.75	10	17.5
Clemas	1.29	15	19.35
Multímetro Digital	12.95	1	12.95
Interruptor Magneto 2M P+N 25A	4.49	5	22.45

**Subtotal acumulado (€)**

2130.44

**Construcción e instalación del sistema.**

Concepto	Precio Ud. (€)	Cantidad	Subtotal (€)
Estructura en hierro para vidrieras	125	4	500
Estructura en hierro para el rosetón	350	1	350
Raíles para las vidrieras delanteras	60	2	120
Montaje de las estructuras	200	1	200
Impresión gráfica de vidrieras	500	4	2000
Impresión gráfica del rosetón	380	1	380
Maderas traseras de vidrieras	30.9	4	123.6
Maderas traseras de rosetón	84	1	84
Broca de metal 3mm	1.95	5	9.75
Broca de metal 4mm pro	2.95	5	14.75
Tornillos 3.5x16	1.95	30	58.5
Tornillos 3.5x19	1.95	20	39
Tornillos 3.5x25	1.95	10	19.5
Cinta de doble cara	5.89	3	17.67

<b>Subtotal acumulado (€)</b>	<b>3916.77</b>
-------------------------------	----------------

<b>Presupuesto total</b>	<b>6047.21 €</b>
--------------------------	------------------

# Bibliografía

- [1] A. Hernández and V. Morales, "Ocio, tiempo libre y animación sociocultural," 2005.
- [2] S. Gorbeña Etxebarria, "Ocio y salud mental."
- [3] Endesa, "Sistemas de iluminación, s.XXI." [Online]. Available: [http://www.endesaeduca.com/Endesa\\_educa/recursos-interactivos/el-uso-de-la-electricidad/xxii.-sistemas-de-iluminacion](http://www.endesaeduca.com/Endesa_educa/recursos-interactivos/el-uso-de-la-electricidad/xxii.-sistemas-de-iluminacion).
- [4] J. Luis, "ILUMINACIÓN CON LEDS," 2014.
- [5] E. Di Bernardo, *Taller de Materialidad II Proyecto de Iluminación Artificial*. 2011.
- [6] McGraw-Hill, *Luminotecnia. Dispositivos para alumbrado incandescente y fluorescente*. .
- [7] Universidad de Buenos Aires, *La fuentes de luz*. 2000.
- [8] Prof. Esteban Sanchis / Prof. Juan B. Ejea, "El diodo de emisión de luz (LED)," pp. 1-37.
- [9] V. D. Lcc and B. Schrank, "Color Theory Color Theory Lecture Topics."
- [10] Revista Española de Electrónica, "Control Inteligente de iluminación LED." [Online]. Available: <http://www.redeweb.com/articulos/articulo.php?id=1154&categoria=componentes>.
- [11] C. H. Dorremochea, M. J. Martorell, and O. S. Jáuregui, "La iluminación con led y el problema de la contaminación lumínica."
- [12] A. G. Smith, *Introduction to Arduino*. 2011.
- [13] M. R. Guti, *Manual de Programación Arduino Arduino : Manual de Programación*. .
- [14] J. P. Dignani, "Análisis del protocolo zigbee," 2011.
- [15] Anónimo, "BLUETOOTH ¿Qué significa Bluetooth?" [Online]. Available: <http://www.monografias.com/trabajos-pdf3/bluetooth-significado/bluetooth-significado.pdf>.
- [16] Rabbit & Dinamic C, *An Introduction to Wi-Fi*. .
- [17] A. A. Akintola, G. A. Aderounmu, A. A. Owojori, M. O. Adigun, and S. Africa, *Performance Modeling of UDP over IP-Based Wireline and Wireless Networks*, vol. 3. 2006.

- [18] NodeMCU Team, "NodeMCU," 2014. [Online]. Available: [http://nodemcu.com/index\\_en.html](http://nodemcu.com/index_en.html).
- [19] Diario Electrónico, "Controladores Básicos (Drivers)," 2013. [Online]. Available: <http://www.diarioelectronicohoy.com/blog/controladores-basicos-drivers>.
- [20] Anónimo, "Transistores." [Online]. Available: <http://www.prometec.net/transistores/>.
- [21] Juan González Gómez (Universidad Pontifica de Salamanca), *Capítulo 6 Decodificadores y Comparadores*. .
- [22] EcuRed, "Tecnología Wi-Fi." [Online]. Available: [http://www.ecured.cu/Tecnolog%C3%ADa\\_Wi-Fi](http://www.ecured.cu/Tecnolog%C3%ADa_Wi-Fi).
- [23] M. A. Casillas, "Topología de redes." [Online]. Available: <http://redestipostopologias.blogspot.com.es/2009/03/topologia-de-redes.html>.
- [24] Universidad de Oviedo, "El protocolo TCP/IP."
- [25] Instituto Nacional de Ciberseguridad de España, "Protege tu WiFi." [Online]. Available: <https://www.osi.es/es/protege-tu-wifi.html>.
- [26] Arduino, "Language Reference." [Online]. Available: <https://www.arduino.cc/en/Reference/HomePage>.
- [27] M. Miller, "What does 'library' mean?" [Online]. Available: <https://www.quora.com/What-does-library-mean-in-the-case-of-programming-languages>.
- [28] W3Schools, "HTML Introduction." [Online]. Available: [http://www.w3schools.com/html/html\\_intro.asp](http://www.w3schools.com/html/html_intro.asp).
- [29] J. E. Pérez, "CSS.", 2013

# APÉNDICES

Código, Proceso de Construcción y  
Esquemático de NodeMCU ESP8266

# A1. Código

Se advierte que el código fuente desarrollado para el proyecto se encuentra en el Repositorio del Departamento de Ingeniería de Automática, Ing. Eléctrica y Electrónicas e Informática Industrial.

## A2. Proceso de construcción

Se presenta el siguiente reportaje fotográfico con la intención de ilustrar el proceso que se ha seguido para la construcción de una de las vidrieras del proyecto.

**Fotografía 1-A2.**



Se puede visualizar la estructura de las vidrieras en hierro. Sus dimensiones son 1,20m en la base y 4,15m de altura.

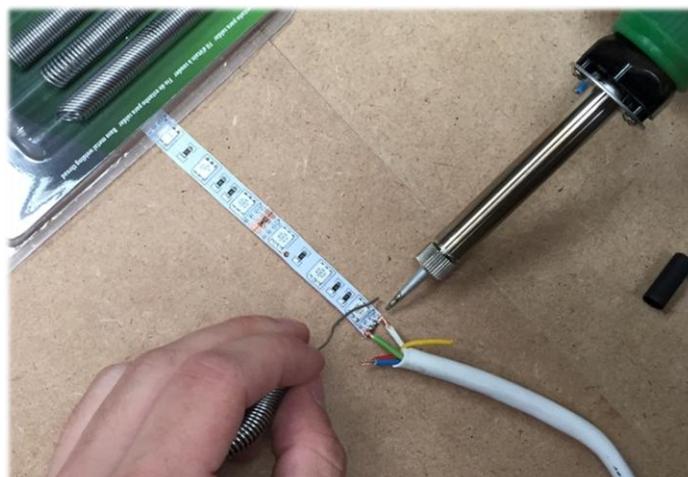
Sin embargo, en la presente fotografía, debido a la perspectiva, no se puede apreciar el tamaño real de la estructura.

Como se puede ver, se trata de una estructura esbelta y estrecha que, tras el ensamblaje de todas sus partes, aparenta ser mucho más robusta.

Los planos acotados se facilitaron al técnico que procedió a la construcción de la misma. Las uniones entre los tubos de hierro se realizaron con soldaduras.

**Fotografía 2-A2.**

Para adaptar las longitudes de las tiras LED a los contornos de los diferentes paneles, es necesario cortar las tiras y posteriormente soldar el cableado a las mismas a través de las zonas destinadas a ello. Para afianzar estas conexiones, se recurrió al uso de tubo termorretráctil.



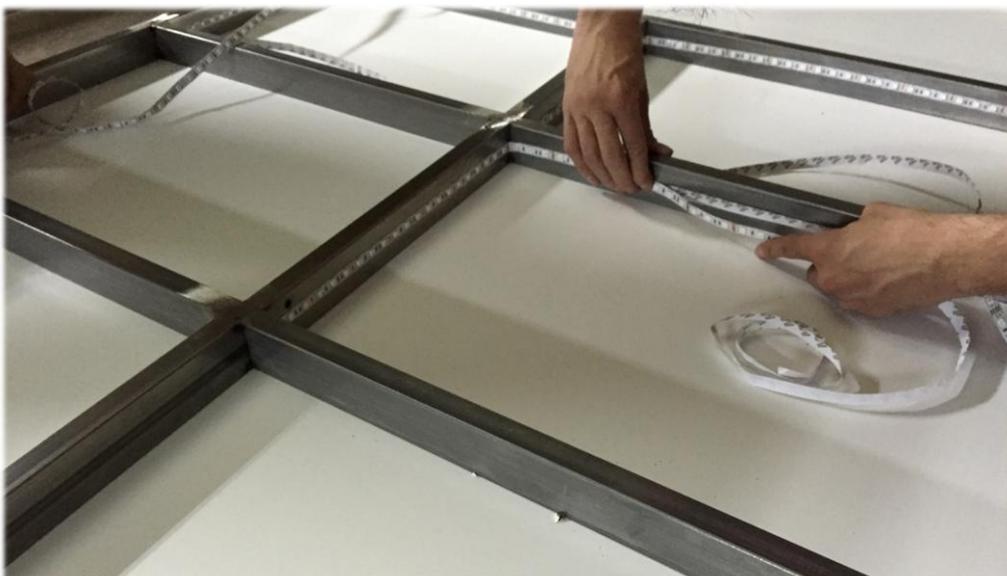
**Fotografías 3,4 -A2.**



A través de las fotografías 3 y 4 se pretende mostrar las vistas frontal y posterior de las estructuras tras la instalación de las maderas traseras. Como puede verse en la figura de la izquierda, el fondo es blanco para actuar como reflector homogeneizando la intensidad lumínica.

**Fotografía 5-A2.**

La instalación de las tiras LED RGB se realiza contorneando los tubos que conforman cada panel. Al contar con un adhesivo, este paso es relativamente sencillo. Sin embargo, si no se tratan con cuidado, las tiras pueden sufrir daños en sus circuitos internos y dejar de lucir. Una vez colocadas, los cables del extremo se pasan a través de un taladro a la superficie trasera.

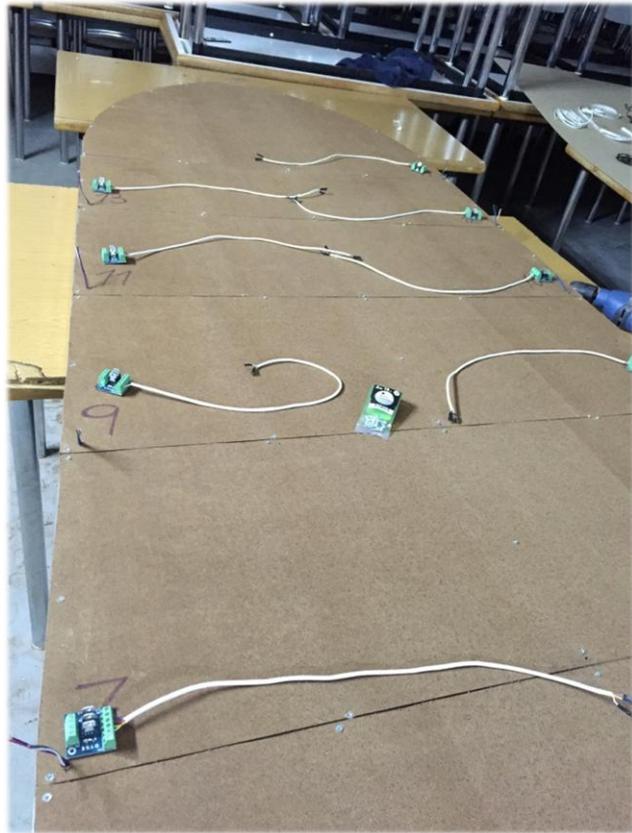


### Fotografía 6-A2.

Instalación de los primeros cables. Como se explica en la Memoria, las tapas traseras de las vidrieras y el rosetón sirve como superficie base para sus circuitos.

En la fotografía se pueden ver los drivers, distribuidos a lo largo de la vidriera. Se encuentran posicionados detrás del panel que alimentan. A través de esta organización, se consigue facilitar la detección de los drivers defectuosos.

Los cables que se dirigen al eje central son las conexiones de control entre los drivers y los módulos WiFi, que se situarán en dicho eje.



### Fotografía 7-A2.

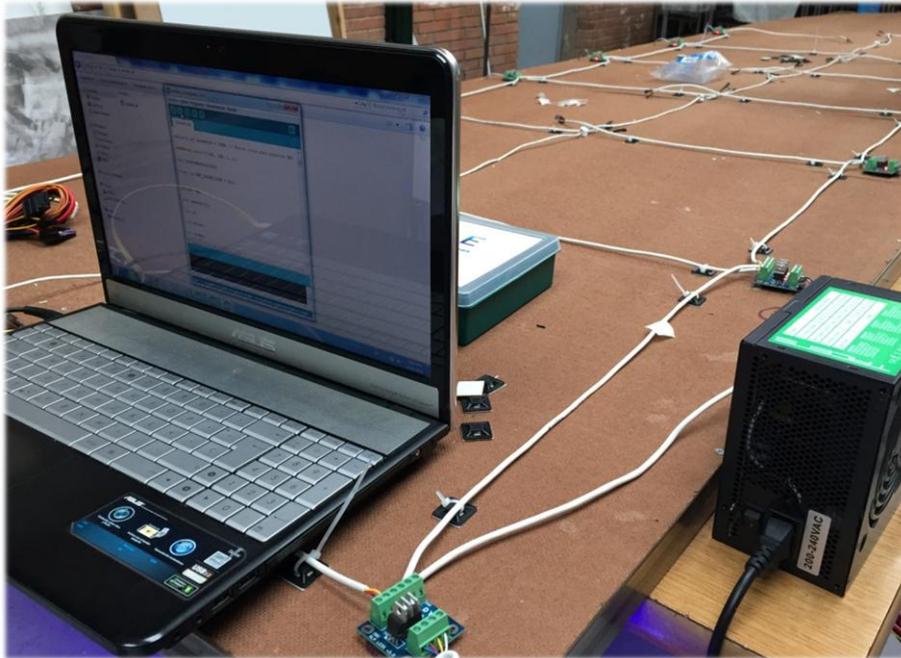
En la fotografía 7 se puede apreciar el cableado que va adherido a la superficie trasera de la vidriera. Además, ya se ha instalado la fuente de alimentación para PC.

El siguiente paso es colocar los módulos esclavos conforme a los circuitos diseñados tras su programación.

Comprobar los contactos de todas las conexiones es fundamental para evitar fallos debidos a la alimentación de los componentes. Una herramienta realmente útil para realizar estas comprobaciones son los Multímetros.



**Fotografía 8-A2.**

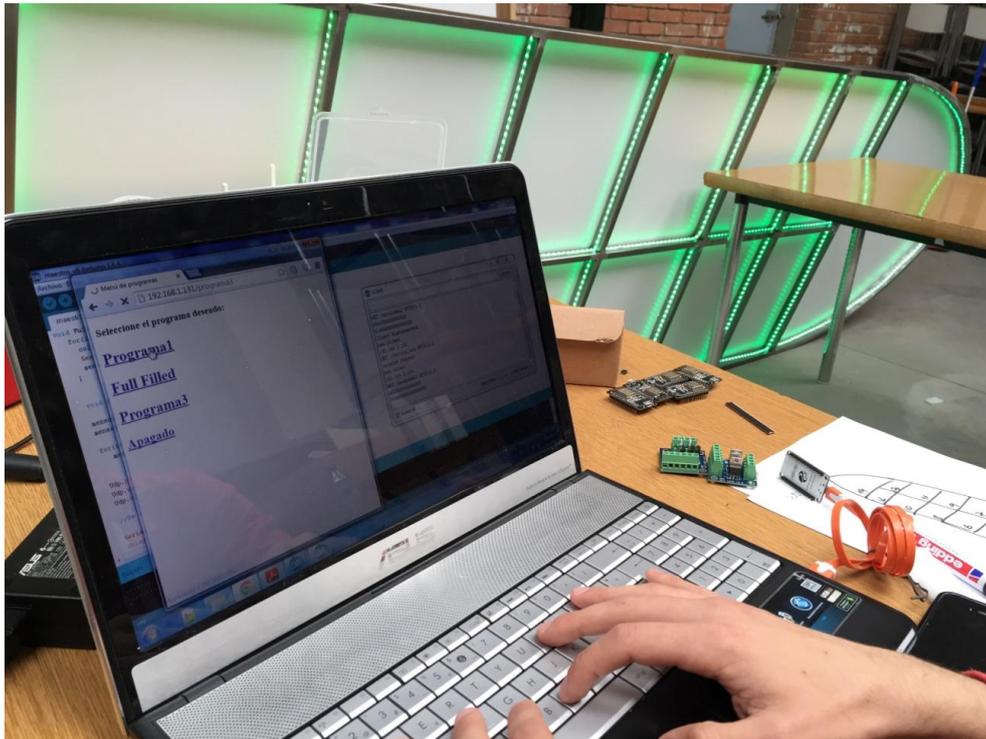


**Fotografía 9-A2.**

En la imagen se muestra el resultado de una de las primeras pruebas. En estas primeras pruebas se trata de iluminar todos los paneles con el mismo color, y probar los LED rojos, verdes y azules. Sin embargo, hasta que se logró encender todas las tiras con los colores deseados, hubo que revisar y corregir muchos errores en la ejecución de los circuitos. La ventaja es que, tras la primera vidriera, la instalación del resto de elementos resulta mucho más sencilla.



**Fotografía 10-A2.**



Una vez queda asegurada la calidad de todas las conexiones y, por tanto, la vidriera se ilumina del modo que se le ordena, se puede proceder a ponerla de pie. Entonces se coloca el material difusor y, rápidamente pueden verse mejoras en la luminosidad de los paneles.

**Fotografías 11, 12, 13 y 14-A2.**



Fotografía 15-A2.



# A3. Esquemático de NodeMCU ESP8266

