

IPN
ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y ELÉCTRICA
UNIDAD CULHUACAN

TESIS INDIVIDUAL

Que como prueba escrita de su Examen Profesional para obtener el Título de **INGENIERO EN COMUNICACIONES Y ELECTRONICA** deberá desarrollar el C.:

CARLOS ALBERTO HUERTA SOLIS
ALEJANDRO MENDOZA LEON

“SISTEMA PARA CONTROLAR FUNCIONES COMPLEMENTARIAS EN UNA BICICLETA A TRAVES DE UNA APLICACION PARA SMARTPHONE”

Actualmente, los problemas generados debido a la enorme densidad demográfica con la que cuenta la Ciudad de México, tales como la movilidad y la contaminación, requiere urgentemente la elaboración de estrategias que contribuyan a solucionarlos.

En ese ámbito, la bicicleta es un vehículo con múltiples particularidades que proporcionan ciertas ventajas con respecto a los medio de transporte convencionales, principalmente la disminución del congestionamiento vial y la nula emisión de gases contaminantes del aire.

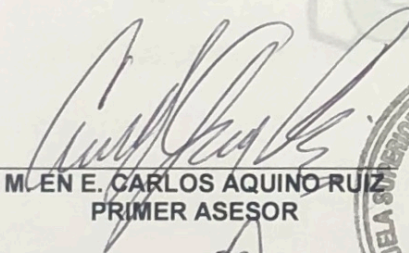
Sin embargo, su rol dentro de la sociedad actual se ve afectado en gran medida debido a la falta de seguridad y la austeridad que conlleva el uso de la bicicleta, aunado a la falta de cultura vial.

Un sistema que optimice el funcionamiento de la bicicleta a través del uso de la tecnología, aportando considerables mejoras en aspectos específicos como la seguridad vial, cuidado ambiental, servicio de geolocalización y evaluación del rendimiento físico; tendrá como resultado un vehículo práctico, ecológico y seguro que incentivará el uso de medios de transporte alternativos en la CDMX.

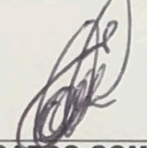
Capitulado:

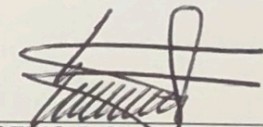
OBJETIVOS
CAPITULO I ESTADO DEL ARTE
CAPITULO II MARCO TEORICO
CAPITULO III DISEÑO E IMPLEMENTACION
CAPITULO IV PRUEBAS Y RESULTADOS
CONCLUSIONES

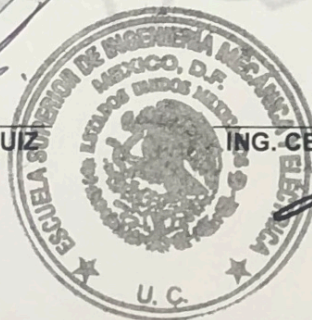
México D. F., a 12 de diciembre del 2018.


M. EN E. CARLOS AQUINO RUIZ
PRIMER ASESOR


ING. CELEDONIO ENRIQUE AGUILAR MEZA
SEGUNDO ASESOR


ING. EDGAR RICARDO GOMEZ NAVARRO
JEFE DE LA CARRERA DE I.C.E


DR. EUSEBIO RICARDEZ VAZQUEZ
SUBDIRECTOR ACADEMICO INTERINO





INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y
ELÉCTRICA
UNIDAD CULHUACAN
SUBDIRECCIÓN ACADÉMICA
OFICINA DE TITULACIÓN PROFESIONAL



CARTA AUTORIZACION DE USO DE OBRA

En la Ciudad de México, a **12 de diciembre del año 2018**, los que suscriben **CARLOS ALBERTO HUERTA SOLIS** y **ALEJANDRO MENDOZA LEON** alumnos de la carrera de **Ingeniería en Comunicaciones y Electrónica**, con número de registro **R-056/18**, egresados de la Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Culhuacan, manifestamos que somos los autores intelectuales del presente trabajo de **Tesis Colectiva**, bajo la asesoría del **M. en E. Carlos Aquino Ruiz** y el **Ing. Celedonio Enrique Aguilar Meza** y otorgamos el uso del trabajo titulado **Sistema para controlar funciones complementarias en una bicicleta a través de una aplicación para Smartphone**, al Instituto Politécnico Nacional, para su difusión con fines académicos y de investigación.

Los usuarios de la información no deberán reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso de los autores y/ o asesores del trabajo. Este puede ser obtenido escribiendo a las siguientes direcciones de correo: huerta-90@hotmail.com y alex.ml816@gmail.com y Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

CARLOS ALBERTO HUERTA SOLIS

ALEJANDRO MENDOZA LEON



INSTITUTO POLITÉCNICO NACIONAL
Escuela Superior de Ingeniería Mecánica y Eléctrica

**SISTEMA PARA CONTROLAR FUNCIONES
COMPLEMENTARIAS EN UNA BICICLETA A TRAVÉS
DE UNA APLICACIÓN PARA SMARTPHONE**

TESIS

**QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMUNICACIONES Y ELECTRÓNICA**

PRESENTAN:

CARLOS ALBERTO HUERTA SOLÍS

ALEJANDRO MENDOZA LEÓN

Asesores: M. en E. Carlos Aquino Ruiz

Ing. Celedonio Enrique Aguilar Meza



**CIUDAD DE MÉXICO
OCTUBRE DEL 2019**



AGRADECIMIENTOS



A mi familia, que con su amor y apoyo incondicional han sido los principales cimientos para la construcción de mi vida personal y profesional.

A mi papá Jesús por sembrar en mí las bases de responsabilidad y los deseos de superación.

A mi mamá Matilde por enseñarme los valores de humildad, sacrificio y espíritu de servicio.

A mis hermanos Jesús y Jacque por estar siempre conmigo, en las buenas y en las malas.

A mis sobrinos Lalo, Emy y Eder por ser siempre una fuente de inspiración y alegría.

A mi institución y a mis maestros, especialmente a nuestros asesores Carlos y Celedonio, que nos han brindado su tiempo, consejo, conocimiento y experiencia.

A mis amigos, que han dado sentido a este camino a través de su apoyo y compañía, en especial a ti Alejandro, por tenderme la mano y confiar en mí en este proyecto.

Carlos Alberto Huerta Solís

A Dios por permitirme concluir de forma satisfactoria una etapa más en mi vida.

A mis padres por inculcarme buenos valores, ayudarme a ser una buena persona, el hábito del estudio y apoyarme en todos los niveles escolares, sin su apoyo este logro no sería posible.

A mi papá, tus sabios consejos me guiaron en este camino siempre tuviste las palabras adecuadas para motivarme y darme fuerzas. Me apoyaste en todas las formas y recursos que tenías, espero que este logro refleje todo el bien que me has hecho, te amo: Elías.

A mi mamá, la dedicación y atención que siempre tuviste conmigo se ven reflejados aquí, el amor y la pasión con la que me veías forjaron en mi hábitos y aptitudes que han hecho de mí una mejor persona. Este logro también es tuyo, te amo: Teresa.

A mi hermano Daniel, has sido un ejemplo a seguir y una meta a superar, me apoyaste para desarrollarme en ámbitos donde no era tan hábil, tus consejos tan certeros en decisiones importantes.

A ti Carlos, por tu amistad, tu dedicación y esmero para poder concluir con este proyecto. Eres una persona noble, honesto y sencillo. Me alegra el haber trabajado contigo.

A los profesores Carlos y Celedonio, por su disposición y orientación. Su apoyo y experiencia nos permitieron darle buen curso al proyecto.

Alejandro Mendoza León



INTRODUCCIÓN



El presente trabajo es resultado del ferviente deseo de contribuir, en la medida de lo posible, al desarrollo de proyectos innovadores y asequibles en el ámbito de la ciencia y la tecnología.

Las horas de esfuerzo y dedicación destinadas a la elaboración de este proyecto tienen como raíz fundamental establecer nuevos caminos a través de los cuales se puede llegar a solucionar problemas, satisfacer ciertas necesidades o, incluso, optimizar el ejercicio de determinadas actividades.

A lo largo de esta tesis, se muestra el desarrollo y evaluación de una aplicación para Smartphone enfocada a la práctica del ciclismo. Se evalúa, a partir de la instalación de un programa (software) dentro de un dispositivo móvil, el rendimiento físico del usuario de una bicicleta durante su recorrido, a partir de la medición y el cálculo de ciertas variables básicas.

De la misma forma, se expone el procedimiento y los elementos electrónicos necesarios para controlar los módulos de iluminación instalados en la bicicleta, exhibiendo también los algoritmos empleados para aprovechar los recursos del dispositivo móvil y utilizarlo como sensor de geolocalización durante el recorrido en bicicleta.

Se tiene plena convicción de que el trabajo que se muestra a continuación está constituido a través de ideas, si bien simples, creativas y, sobre todo, eficientes; surgidas a partir de los conocimientos adquiridos a lo largo de nuestra trayectoria escolar y complementados por un proceso de investigación orientado al rubro de las aplicaciones móviles.

En el primer capítulo se hace referencia al proceso evolutivo de la bicicleta a lo largo del tiempo, desde su aparición hasta nuestros días. Así mismo, se hace un análisis y comparación de aquellos proyectos innovadores recientes que, de alguna u otra forma, han conseguido complementar las funciones de las bicicletas convencionales a través de la implementación de gadgets y a través del uso de apps.

Posteriormente, en el segundo capítulo, se muestran las herramientas de diseño y programación necesarias para el desarrollo de la aplicación móvil, compatible con el sistema operativo Android.

En el capítulo tres se abordan las principales características técnicas de cada uno de los elementos que integran el sistema para bicicleta, así como sus variaciones de uso. Además, se hace un análisis de algunos modelos matemáticos enfocados en



medir la capacidad de las personas en el ámbito del deporte, específicamente en ciclismo.

En la siguiente sección, el tercer capítulo, se explican propiamente los recursos de software y los códigos empleados en el diseño y programación de las diferentes interfaces que conforman la app, desde los bocetos hasta su versión final.

Así mismo, se describe el proceso de transferencia de información entre el dispositivo móvil (Smartphone) y los diversos módulos y dispositivos que conforman el hardware del sistema. Igualmente, se aclaran los algoritmos y modelos matemáticos empleados para la obtención de los parámetros de rendimiento físico.

Por último, se establecen las observaciones y conclusiones obtenidas después de haber realizado las pruebas de campo correspondientes, así como sus respectivas modificaciones y adecuaciones, conforme a cada uno de los objetivos planteados.

Estando conscientes de que no hay mayor satisfacción que la proveniente del trabajo arduo y constante enfocado en hacer realidad las propias ideas y siguiendo los principios de poner la técnica al servicio de la patria, se pone a su disposición este proyecto de investigación, que si bien, no es el más profundo ni ostentoso, propone una solución práctica y eficiente en el ámbito en el cual está destinado.



ÍNDICE DE CONTENIDO

INDÍCE DE CONTENIDO	V
ÍNDICE DE FIGURAS	VII
ÍNDICE DE TABLAS	IX
PLANTEAMIENTO DEL PROBLEMA	X
JUSTIFICACIÓN	XII
OBJETIVO GENERAL	XIV
OBJETIVOS PARTICULARES	XIV
1 ESTADO DEL ARTE.....	- 16 -
1.1 HISTORIA DE LA BICICLETA.....	- 16 -
1.2 EMPRESAS QUE DESARROLLAN APPS Y GADGETS PARA BICICLETAS	- 18 -
1.3 PROYECTOS DE SISTEMAS PARA BICICLETAS.....	- 19 -
1.3.1 BICIWOOD.....	- 19 -
1.3.2 HAMMERHEAD.....	- 20 -
1.3.3 DUBIKE.....	- 21 -
1.3.4 COBI.....	- 22 -
1.3.5 SMART HALO.....	- 23 -
1.4 LÍNEA DEL TIEMPO DE SISTEMAS PARA BICICLETAS	- 24 -
2 MARCO TEÓRICO.....	- 26 -
2.1 SMARTPHONE.....	- 26 -
2.2 APPS.....	- 28 -
2.2.1 Aplicaciones Nativas.....	- 28 -
2.2.2 Aplicaciones Web	- 29 -
2.2.3 Aplicaciones Híbridas.....	- 29 -
2.3 SOFTWARE DE DISEÑO	- 30 -
2.4 ARDUINO.....	- 34 -
2.4.1 Arduino UNO.....	- 34 -
2.5 MÓDULO BLUETOOTH	- 35 -
2.6 RELEVADOR	- 36 -
2.7 LUCES.....	- 36 -
2.8 SENSOR DE PULSO CARDIACO.....	- 38 -
2.9 ESTIMACIÓN DE RENDIMIENTO FÍSICO	- 40 -
2.10 FRECUENCIA CARDIACA.....	- 39 -
2.11 DIAGRAMA A BLOQUES DEL SISTEMA PARA BICICLETA.....	- 40 -



3	DISEÑO E IMPLEMENTACIÓN.....	- 42 -
3.1	DESARROLLO DE LA APP	- 42 -
3.2	CONEXIÓN BLUETOOTH	- 55 -
3.3	SKETCH ARDUINO	- 64 -
3.4	SISTEMA DE ILUMINACIÓN.....	- 64 -
3.5	GRAFICA DE PULSO CARDÍACO.....	- 64 -
3.6	MÓDULO DE RELEVADORES.....	- 64 -
3.7	MÓDULO DE CONTROL.....	- 64 -
3.8	CÁLCULO DE PARÁMETROS DE RENDIMIENTO FÍSICO.....	- 64 -
3.9	DIAGRAMA DE FLUJO	- 67 -
4	PRUEBAS Y RESULTADOS.....	- 70 -
4.1	APP.....	- 70 -
4.1.1	<i>Descarga de la aplicación móvil</i>	<i>- 70 -</i>
4.1.2	<i>Ícono y pantalla de inicio</i>	<i>- 72 -</i>
4.1.3	<i>Registro y validación de usuario.....</i>	<i>- 73 -</i>
4.1.4	<i>Geolocalización, distancia, tiempo y velocidad</i>	<i>- 77 -</i>
4.1.5	<i>Control de luces.....</i>	<i>- 79 -</i>
4.1.6	<i>Parámetros de rendimiento físico.....</i>	<i>- 84 -</i>
4.1.7	<i>Interfaces que componen la App.....</i>	<i>- 84 -</i>
4.2	PRUEBAS E INSTALACIÓN DEL SISTEMA.....	- 87 -
	CONCLUSIONES.....	- 90 -
	ANEXOS	- 92 -
	GLOSARIO	- 99 -
	BIBLIOGRAFÍA	- 102 -



ÍNDICE DE FIGURAS

1 ESTADO DEL ARTE

FIGURA 1. 1 PROTOTIPO HECHO EN FUNCIÓN DE LOS SUPUESTOS PLANOS DE DA VINCI	- 16 -
FIGURA 1. 2 DISEÑO DE BICICLETA DE KARL DREIS LLAMADO "DRASIANA"	- 17 -
FIGURA 1. 3 MODELO "SAFETY BICYCLE" DE 1885 CREADO POR JOHN KEMP STARLEY	- 17 -
FIGURA 1. 4 APPS SOBRESALIENTES PARA CICLISTAS	- 18 -
FIGURA 1. 5 BICICLETA DE BICIWOOD.....	- 19 -
FIGURA 1. 6 HAMMERHEAD MONTADO EN UN MANUBRIO DE BICICLETA	- 20 -
FIGURA 1. 7 VISTA DE MANUBRIO DE DUBIKE.....	-21-
FIGURA 1. 8 VISTA TRASERA DE DUBIKE.....	- 21 -
FIGURA 1. 9 MÓDULOS DEL SISTEMA PARA BICICLETA COBI	- 22 -
FIGURA 1. 10 GADGET SMARTHALO Y SU APP CORRESPONDIENTE	- 23 -
FIGURA 1. 11 LÍNEA DEL TIEMPO DE PROYECTOS CON SISTEMAS PARA BICICLETAS	- 24 -

2 MARCO TEÓRICO

FIGURA 2. 1 APLICACIÓN NATIVA EN DISPOSITIVO NOKIA	- 28 -
FIGURA 2. 2 APLICACIÓN WEB EN DISPOSITIVO APPLE	- 29 -
FIGURA 2. 3 APLICACIÓN HÍBRIDA EN DISPOSITIVOS CON SISTEMA OPERATIVO IOS Y ANDROID	- 30 -
FIGURA 2. 4 DESARROLLO DE UNA APP A TRAVÉS DE LA PLATAFORMA MIT APP INVENTOR 2	- 31 -
FIGURA 2. 5 PLATAFORMA DE DISEÑO DE MIT APP INVENTOR 2.....	- 32 -
FIGURA 2. 6 EDITOR DE BLOQUES DE MIT APP INVENTOR 2	- 33 -
FIGURA 2. 7 ARDUINO UNO	- 34 -
FIGURA 2. 8 MÓDULO BLUETOOTH HC-05	- 35 -
FIGURA 2. 9 RELEVADOR ELECTROMAGNÉTICO	- 36 -
FIGURA 2. 10 LUZ URBANA RURAL.....	- 36 -
FIGURA 2. 11 LUZ LED TRASERA.....	- 37 -
FIGURA 2. 12 SENSOR DE PULSO CARDÍACO	- 40 -
FIGURA 2. 13 DIAGRAMA A BLOQUES DEL SISTEMA PARA BICICLETA	- 40 -

3 DISEÑO E IMPLEMENTACIÓN

FIGURA 3. 1 LISTA DE PROYECTOS CREADOS EN UNA CUENTA DE MIT APP INVENTOR 2	- 43 -
FIGURA 3. 2 PLATAFORMA DE DISEÑO DE APP EN MIT APP INVENTOR 2	- 43 -
FIGURA 3. 3 PLATAFORMA DE BLOQUES DE APP EN MIT APP INVENTOR 2.....	- 44 -
FIGURA 3. 4 PROCESO DE ELABORACIÓN DEL SPLASH SCREEN O PANTALLA DE INICIO	- 45 -
FIGURA 3. 5 PROCESO DE ELABORACIÓN DE LA PANTALLA DE VALIDACIÓN DE USUARIO	- 46 -
FIGURA 3. 6 PROCESO DE ELABORACIÓN DE INTERFAZ PARA DAR DE ALTA UNA CUENTA	- 47 -
FIGURA 3. 7 PROCESO DE ELABORACIÓN DE LA BASE DE DATOS	- 48 -
FIGURA 3. 8 PROCESO DE ELABORACIÓN DEL MAPA DE GEOLOCALIZACIÓN Y MEDIDOR DE VELOCIDAD	- 49 -



FIGURA 3. 9 PROCESO DE ELABORACIÓN DEL RESUMEN DE RECORRIDO Y MAPA DE RUTA	- 50 -
FIGURA 3. 10 PROCESO DE ELABORACIÓN DE HISTORIAL DE RECORRIDOS	- 51 -
FIGURA 3. 11 PROCESO DE ELABORACIÓN DEL CONTROL DE LUCES	- 52 -
FIGURA 3. 12 PROCESO DE OBTENCIÓN DE PARÁMETROS DE RENDIMIENTO FÍSICO	- 53 -
FIGURA 3. 13 PROCESO DE OBTENCIÓN DE GRÁFICA DE PULSO CARDÍACO	- 54 -
FIGURA 3. 14 SKETCH DESTINADO A ESTABLECER UNA COMUNICACIÓN SERIAL CON LA PC Y EL HC-05	- 55 -
FIGURA 3. 15 DIAGRAMA DE CONEXIÓN DEL MÓDULO HC-05 Y LA PLATAFORMA ARDUINO	- 56 -
FIGURA 3. 16 SKETCH DESTINADO A ESTABLECER UNA COMUNICACIÓN ENTRE PLATAFORMA ARDUINO Y MÓDULO HC-05	- 57 -
FIGURA 3. 17 LISTA DE DISPOSITIVOS BLUETOOTH VINCULADOS CON EL SMARTPHONE	- 59 -
FIGURA 3. 18 PRUEBA DE CONTROL DE ENCENDIDO Y APAGADO DE LOS LEDS MEDIANTE LA PLATAFORMA ARDUINO	- 59 -
FIGURA 3. 19 CIRCUITO DE PRUEBA DE SENSOR DE PULSO CARDIACO	- 60 -
FIGURA 3. 20 GRÁFICA DE PULSO CARDÍACO ARDUINO	- 60 -
FIGURA 3. 21 DIAGRAMA CONTROL DE LUCES A TRAVÉS DE LA PLATAFORMA ARDUINO	- 61 -
FIGURA 3. 22 DIAGRAMA DE MÓDULO DE CONTROL DE SISTEMA PARA BICICLETA	- 62 -
FIGURA 3. 23 CIRCUITO DE MÓDULO DE CONTROL DE SISTEMA PARA BICICLETA	- 63 -
FIGURA 3. 24 PLACA DE CIRCUITO IMPRESO	- 64 -
FIGURA 3. 25 VISTA LATERAL DEL CIRCUITO DE MÓDULO DE CONTROL	- 64 -
FIGURA 3. 26 DIAGRAMA DE FLUJO DEL SISTEMA ECO BICI	- 68 -

4 PRUEBAS Y RESULTADOS

FIGURA 4. 1 COMPILACIÓN Y DESCARGA DE APP DESDE LA PLATAFORMA APP INVENTOR 2	- 71 -
FIGURA 4. 2 ÍCONO DE LA APP Y SPLASH SCREEN	- 72 -
FIGURA 4. 3 PROCESO DE REGISTRO Y VALIDACIÓN DE CUENTA DEL USUARIO	- 73 -
FIGURA 4. 4 COMPONENTES DE LA INTERFAZ "VALIDACIÓN DE USUARIO"	- 74 -
FIGURA 4. 5 COMPONENTES DE LA INTERFAZ "REGISTRO DE USUARIO"	- 75 -
FIGURA 4. 6 COMPONENTES DE LA BASE DE DATOS DE LA APP	- 76 -
FIGURA 4. 7 PROCESO DE GEOLOCALIZACIÓN Y RESUMEN DE RECORRIDO	- 77 -
FIGURA 4. 8 COMPONENTES DE LA INTERFAZ "GEOLOCALIZACIÓN"	- 77 -
FIGURA 4. 9 COMPONENTES DE LA INTERFAZ "GOOGLE MAPS"	- 78 -
FIGURA 4. 10 COMPONENTES DE LA INTERFAZ "RESUMEN DE RECORRIDO"	- 80 -
FIGURA 4. 11 COMPONENTES DEL MENÚ PRINCIPAL DE LA APP	- 81 -
FIGURA 4. 12 PROCESO PARA EL CONTROL DE LUCES DESDE UNA APP	- 82 -
FIGURA 4. 13 COMPONENTES DE LA INTERFAZ "CONTROL DE LUCES"	- 83 -
FIGURA 4. 14 PROCESO PARA DESPLIEGUE DE PARÁMETROS Y GRÁFICA DE PULSO CARDÍACO	- 84 -
FIGURA 4. 15 COMPONENTES DE LA INTERFAZ "PARÁMETROS DE RENDIMIENTO FÍSICO"	- 85 -
FIGURA 4. 16 INTERFACES QUE COMPONEN LA APP "ECO BICI"	- 86 -
FIGURA 4. 17 MÓDULO DE ILUMINACIÓN	- 87 -
FIGURA 4. 18 LUCES TRASERA Y DELANTERA	- 87 -
FIGURA 4. 19 SUJETADOR DE SMARTPHONES PARA BICICLETA	- 87 -
FIGURA 4. 20 PRUEBA DE LUZ TRASERA Y DELANTERA	- 88 -
FIGURA 4. 21 PRUEBA DE APP CPON EL SISTEMA DE ILUMINACIÓN	- 88 -
FIGURA 4. 22 PRUEBA DE SENSOR DE PULSO CARDÍACO	- 88 -
FIGURA 4. 23 ADAPTADOR DE PULSO CARDÍACO	- 88 -
FIGURA 4. 24 INSTALACIÓN DE SISTEMA PARA BICICLETA	- 89 -



ÍNDICE DE TABLAS

1 ESTADO DEL ARTE

TABLA 1. 1 FUNCIONES BÁSICAS DE PROYECTOS CON SISTEMAS PARA BICICLETA - 24 -

2 MARCO TEÓRICO

TABLA 2. 1 TIPOS DE SENSORES SOPORTADOS POR LA PLATAFORMA ANDROID..... - 27 -

3 DISEÑO E IMPLEMENTACIÓN

TABLA 3. 1 COMANDOS PARA LA CONFIGURACIÓN DEL MÓDULO BLUETOOTH HC-05 - 56 -

TABLA 3. 2 RELACIÓN DE PORCENTAJES ENTRE LA FC MÁX EMPLEADA Y EL VO₂ MÁX CONSUMIDO - 66 -

TABLA 3. 3 NIVELES DE VO₂ MÁX (ML/KG.MIN) EN HOMBRES - 66 -

TABLA 3. 4 NIVELES DE VO₂ MÁX (ML/KG.MIN) EN MUJERES - 66 -



PLANTEAMIENTO DEL PROBLEMA



En la actualidad, es evidente en la Ciudad de México y sus alrededores que el uso excesivo del automóvil, aunado al alto grado de sobrepoblación, ha derivado en diversos conflictos que afectan directamente la calidad de vida y la salud de sus habitantes.

En la Ciudad de México circulan alrededor de 5 millones de autos cada día. A ello se suman 2 millones que ingresan desde la zona conurbada. Además, cada año se compran e integran al padrón vehicular unos 200 mil autos más.

Las vías de tránsito no son capaces de satisfacer eficientemente la creciente demanda por parte de los automóviles. Como consecuencia, el congestionamiento vial ya es una constante durante los desplazamientos en el interior de la ciudad.

Según información del periódico La Jornada, en 1994, una persona empleaba en trasladarse hasta su trabajo en promedio 27 minutos, en el año 2007 el promedio ya era de 59 minutos. Para el 2015, el incremento en tiempo ya era de un 300% [1].

Notoriamente la velocidad de circulación se ha visto reducida: Un promedio de 6 kilómetros por hora en lapsos de máxima demanda, que se refleja en una pérdida económica anual de 300 mil millones de pesos de acuerdo con el Instituto de Políticas para el Transporte del Desarrollo. El gasto de transporte de un habitante en la CDMX con automóvil es de entre 100 y 300 pesos diarios.

Por otra parte, los índices de contingencia ambiental en la ciudad son cada vez más preocupantes debido, en parte, a las grandes cantidades de contaminantes emitidos por los motores de combustión interna.

El Reporte Nacional de Movilidad Urbana en México 2014-2015 señala que, los congestionamientos viales en las principales arterias que comunican los centros urbanos conllevan a un deterioro de la calidad del aire y por ende de la salud de la población, ya que las ciudades mexicanas padecen serios problemas de contaminación ambiental, donde el sector transporte es una de sus principales causas, al contribuir con 20.4% de la emisión de gases de efecto invernadero (GEI), de los cuales 16.2% proviene del subsector automotor [2]. Así lo señalaron representantes del Centro Mexicano de Derecho Ambiental AC (Cemda), así como el Instituto Internacional de Transporte Limpio (ICCT), quienes destacaron que México tiene actualmente estándares muy laxos para controlar las emisiones contaminantes que son percursores de ozono [3].



En este contexto, medios de transporte alternativos como la bicicleta podrían ser de gran utilidad debido a sus múltiples beneficios sobre los medios de transporte convencionales. No obstante, es una realidad que su rol dentro de la sociedad actual se ve sobrepasado de forma considerable por un medio que ofrece innegables ventajas en el ámbito de seguridad, comodidad y rapidez, como es el caso del automóvil.

Y es que, según el Instituto Nacional de Estadística, Geografía e Informática (INEGI), en la Ciudad de México hay 352 accidentes de ciclistas al año [4].

La distracción, los giros y adelantamientos incorrectos o la invasión repentina de otro carril son las principales causas de los accidentes en los que están implicados conductores y ciclistas. Quedando así rezagada como una opción muy austera y de alto riesgo, especialmente durante la noche.

Por otra parte, en el mundo actual, el desarrollo tecnológico basado en los dispositivos electrónicos ha tomado un rol fundamental a tal grado de ser considerado la base de la civilización moderna.

Un estudio realizado por la Asociación Mexicana de Internet (AMIPCI) y la Comisión de Agenda Digital y Tecnologías de la Información, concluyó que los usuarios se conectan en promedio 6 horas y 11 minutos al día a Internet.

Las laptops (68%), smartphones (58%) y tabletas (31%) son los dispositivos que han aumentado en la penetración de acceso a Internet. La PC es la única que ha perdido presencia.

Y es que, existe una creciente tendencia dentro de la población en nuestro país que opta por adoptar un estilo de vida sedentario, en gran parte por el uso desmedido de los avances tecnológicos, especialmente los gadgets.

La cifra de la Encuesta Nacional de Nutrición y Salud (Ensanut 2014) es por demás contundente: 56.2% de la población mexicana lleva una vida sedentaria. Cabe mencionar que el sedentarismo está asociado a la aparición y falta de control de diversas enfermedades crónicas como: obesidad, hipertensión, diabetes, mellitus, dislipidemias, osteoporosis y ciertos tipos de cánceres [5].



JUSTIFICACIÓN

El crecimiento poblacional y la necesidad de regular la expansión en la Ciudad de México requieren urgentemente la elaboración de estrategias que contribuyan a solucionar uno de sus más grandes problemas: la movilidad.

La bicicleta, como medio de transporte, presenta múltiples particularidades que proporcionan al usuario ciertas ventajas respecto a los medios de transporte convencionales.

Empezando por su sencillo principio de funcionamiento y su facilidad de manejo, lo que la hace apta para todo tipo de usuarios. Aunado a esto, usar bicicleta es una excelente forma de realizar ejercicio cardiovascular mientras se realiza un desplazamiento de forma relativamente rápida, ya que requiere la activación de los músculos y las articulaciones del cuerpo.

Según un estudio realizado por el C40 sobre las políticas de movilidad peatonal y en bici en la Ciudad de México, una persona que deja el auto y opta por la bicicleta como medio de transporte, puede ganar de tres a cuatro meses de vida por el aumento de actividad física.

Igualmente, es un medio muy práctico a causa de su ligera estructura, lo que se traduce en una mayor capacidad de tránsito dentro de una determinada vía con respecto a los otros medios de transporte y consecuentemente en un menor congestionamiento vial. Un auto ocupa el lugar de hasta 30 bicicletas en circulación y el lugar de 18 bicicletas estacionadas.

Otro aspecto importante que considerar es que no requiere el gasto de combustibles fósiles para su uso, debido a que su funcionamiento es meramente mecánico, lo que la convierte en un vehículo amigable con el ambiente, evitándose así las emisiones de contaminantes del aire.

Según cifras del Instituto para el Transporte y el Desarrollo de Políticas (ITDP) en conjunto con la Universidad de California Davis, el incremento del uso de la bicicleta como medio de transporte podría cortar las emisiones de Dióxido de Carbono un 11% en un lapso de 35 años en la Ciudad de México [6].

Por otra parte, el apogeo de los avances tecnológicos ha dado como resultado toda una gama de dispositivos inteligentes, Smartphones, capaces de realizar una gran diversidad de funciones que pueden ser aprovechados casi en cualquier ámbito de la vida cotidiana, incluyendo los medios de transporte. Todo esto simplemente con hacer uso de determinadas aplicaciones muy fáciles de manejar para el usuario.



Por todo lo anterior, vale la pena replantearse los alcances que puede llegar a tener el uso la bicicleta como medio de transporte en caso de complementar su funcionamiento a través del uso de los Smartphones y otros avances tecnológicos que, al ser implementados en ella, puedan contribuir a tener un recorrido más ameno y reducir los inconvenientes presentados en las bicis convencionales.

Y es que, en un mundo integrado por la información, no pueden ignorarse las mejores prácticas en cuestión de movilidad, como las que hace varios años se desarrollan en países como Holanda, Dinamarca o Alemania, donde el uso de la bicicleta supone, respectivamente, el 28%, 20% y el 12% de los desplazamientos urbanos. En Asia, especialmente en China e India, la bicicleta es el principal medio de transporte.

Considerando lo anterior es posible inferir que, un sistema que optimice el funcionamiento de una bicicleta a través de una sencilla interfaz, aportando considerables mejoras en aspectos específicos como seguridad vial, cuidado ambiental, servicio de geolocalización y evaluación del rendimiento físico, tendrá como resultado un medio de transporte práctico, ecológico y confortable que estará por encima de los vehículos convencionales en las preferencias de los usuarios en múltiples situaciones de la vida cotidiana dentro de la Ciudad de México.



OBJETIVO GENERAL

Diseñar e implementar un sistema para controlar funciones complementarias en una bicicleta a través de una aplicación para Smartphone.

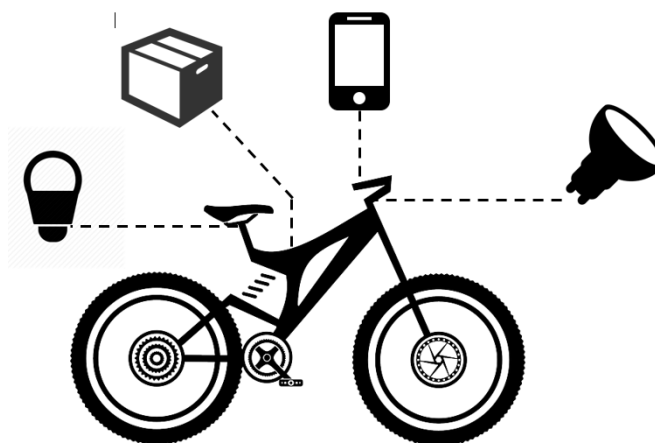
OBJETIVOS PARTICULARES

Desarrollar una aplicación para Smartphone compatible con el sistema operativo Android que sea para el usuario una interfaz interactiva a través de la cual sea posible controlar el módulo de iluminación, geolocalización y rendimiento físico.

Diseñar e implementar un sistema de iluminación frontal y posterior de la bicicleta, activado a través de un módulo Bluetooth compatible con la aplicación para Android.

Diseñar e implementar una base de datos con información del usuario con la finalidad de obtener y mostrar, gráfica y numéricamente, una estimación de su rendimiento físico durante su recorrido actual y el historial de estos.

Implementar un mapa de geolocalización basado en Google Maps con el objetivo de registrar y mostrar la ubicación, distancia, velocidad promedio y tiempo del recorrido actual.





CAPÍTULO 1



ESTADO DEL ARTE

1

ESTADO DEL ARTE

En el presente capítulo se hace referencia al proceso evolutivo de la bicicleta a lo largo el tiempo, desde su aparición hasta nuestros días. Así mismo, se hace un estudio y comparación de aquellos proyectos innovadores que, de alguna forma, han conseguido complementar las funciones de las bicicletas convencionales a través de la implementación de gadgets y a través del uso de apps.

Por medio de este análisis, se pretende obtener un punto de convergencia de las necesidades más importantes para los usuarios de las bicicletas que pueden ser resueltas a través de medios tecnológicos.

1.1 HISTORIA DE LA BICICLETA

La bicicleta, es un vehículo de transporte personal de propulsión humana. Sus componentes básicos son dos ruedas, un sistema de transmisión de pedales, un cuadro que le da la estructura e integra los componentes, un manillar para controlar la dirección y un asiento o sillín para sentarse.

Existe la creencia de que Leonardo Da Vinci creó el primer boceto en papel de una bicicleta a finales del siglo XV, este prototipo fue elaborado de madera como lo muestra la *figura 1.1*.



Figura 1.1 Prototipo hecho en función de los supuestos planos de Da Vinci

En 1816, Karl Dreis, inventor y aristócrata alemán creó lo que sería el primer vehículo dirijible con dos ruedas en línea. Éste tiene el diseño de la *figura 1.2* y no contaba con pedales, cadenas, ni frenos, sino que avanzaba y frenaba con los pies, pero sí tenía las ruedas, sillín, manillar, cuadro y un sistema de dirección bastante más aparatoso que el que hoy conocemos.



Figura 1. 2 Diseño de bicicleta de Karl Dreis llamado "Drasiana"

En 1839, Kirkpatrick Macmillan, un herrero de origen escocés añadió pedales con barras a un prototipo. Estas innovaciones permitieron al ciclista impulsar la máquina con los pies sin tocar el suelo. El mecanismo de impulsión consistía en pedales cortos fijados a la rueda trasera y conectados por barras a unos pedales situados delante del ciclista. Era impulsada por el empuje de los pies hacia abajo y hacia adelante.

En 1885, el británico John Kemp Starley crea la "bicicleta de seguridad" o *Safety Bicycle*, muy parecida a una bicicleta urbana actual como se puede apreciar en la *figura 1.3*. Tenía frenos y la postura era mucho más cercana al suelo, de ahí su nombre. Se añadieron poco después, en 1888, los neumáticos con cámara de aire desarrollados por el irlandés John Boyd Dunlop [7].



Figura 1. 3 Modelo "Safety Bicycle" de 1885 creado por John Kemp Starley

En la actualidad, existen gran variedad de vehículos, considerados como bicicletas, con diferentes características y enfoques dependiendo del tipo de uso que se les dé. Dichas particularidades son de gran importancia para los usuarios debido a que representan la adaptación de su estilo y su forma de vida dentro de su entorno a través de la bicicleta y facilitan su control sobre la misma.

1.2 EMPRESAS QUE DESARROLLAN APPS Y GADGETS PARA BICICLETAS

Hoy en día el uso de las apps para dispositivos inteligentes y gadgets es una tendencia en todos los ámbitos, y el ciclismo no es la excepción. En los últimos años se han desarrollado diversas aplicaciones y dispositivos que posibilitan al usuario tener nuevas experiencias de uso, desarrollándose así diversos sistemas.

Distintas apps para Smartphone se han destacado en cuanto a la interacción entre el usuario y el entorno a través de la bicicleta, algunas de las más sobresalientes son las siguientes [8]:



Figura 1. 4 Apps sobresalientes para ciclistas

Por otra parte, los innegables avances en el ámbito tecnológico han llevado la evolución de las bicicletas más allá de su mera interacción con aplicaciones móviles. Recientemente, en diversas partes del mundo, se han puesto en marcha serios proyectos que amenazan con inundar el mercado en un futuro no muy lejano, ya que llevan la experiencia del ciclismo más allá de los límites, al mezclarla con distintos tipos de gadgets.

Ante la creciente demanda de soluciones tecnológicas en el ámbito del ciclismo, empresas de todo el mundo como Bamboo Tec, Bai Du, Cobi, entre otras, respaldan e invierten hoy en día el desarrollo de este tipo de proyectos.

Cabe mencionar que, los proyectos realizados por las empresas anteriormente citadas, a pesar de haber sido desarrollados de forma independiente, cuentan con elementos en común.

Por ende, el análisis de las principales características de dichos proyectos ha de mostrar la tendencia de las necesidades más relevantes en la rama del ciclismo. Por esta razón, se muestran a continuación los proyectos que en los últimos tiempos han tomado mayor relevancia a nivel internacional, teniendo como elemento nuclear la bicicleta.

1.3 PROYECTOS DE SISTEMAS PARA BICICLETAS

1.3.1 BICIWOOD

Es un proyecto de manufactura 100% mexicana realizado por alumnos de la Universidad del Valle de Mérida, Yucatán. Consiste en una bicicleta constituida en su mayoría por bambú, específicamente bambú hindú. Esta bicicleta cuenta con un dispositivo electrónico que aprovecha la energía producida al pedalear y la convierte a energía eléctrica para poder almacenarla y con esto cargar algún celular o cualquier otro dispositivo vía USB.

La transferencia de energía es posible gracias a un convertidor, compuesto de un circuito instalado en la llanta. Una vez transformada en electricidad, viaja debajo del asiento donde la parte central del sistema regula el voltaje y previene descargas. Un regulador se encarga de administrar la carga eléctrica de manera uniforme, permitiendo al ciclista movimientos irregulares, como acelerar o frenar, sin consecuencias.

Una pantalla LCD muestra datos estadísticos como la velocidad a la que se mueve la bicicleta, la distancia recorrida, energía producida e incluso calorías quemadas.

El marco de la bicicleta está fabricado a partir de palos de bambú unidos mediante una resina especial como se muestra en la *figura 1.5*. Puede soportar hasta 120Kg y es mucho más ligera que un modelo convencional [9].



Figura 1. 5 Bicicleta de BICIWOOD

1.3.2 HAMMERHEAD

Hammerhead es un gadget que al conectarlo a un teléfono iPhone o Android, genera rutas ciclistas en la zona en la que se encuentre el usuario, las cuales se adaptan a sus preferencias (duración, dificultad, paisajismo, entre otras).

El dispositivo está hecho de tal forma que no sea necesario detenerse para comprobar la ruta o consultar la pantalla del móvil. Los LEDs en el Hammerhead dirigen al usuario hasta su destino a través de señales luminosas, como se puede observar en la *figura 1.6*.

Cuenta con un sencillo diseño en forma de T, y se puede instalar en cualquier manillar, permitiendo que el ciclista no pierda en ningún momento el control de la bicicleta. Las luces, además de guiar e indicar la ruta más fácil, facilitan al ciclista ser advertido por el resto de los usuarios de la vía gracias a su particular iluminación.

Hammerhead es compatible con iOS y Android, permitiendo al usuario disfrutar de las muchas aplicaciones de los smartphones. La aplicación asociada al dispositivo permite crear nuevas rutas y buscar recorridos en función de la distancia que se pretende recorrer, el paisaje, el tiempo disponible o incluso, la presencia de cuevas u otras características del terreno [10].



Figura 1. 6 Hammerhead montado en un manubrio de bicicleta

1.3.3 DUBIKE

Dubike es una bicicleta que cuenta con sensores que registran información acerca de las constantes vitales y actividad del usuario, por ejemplo: ritmo de pedaleo, frecuencia cardíaca o incluso la presión ejercida sobre los pedales. Toda esta información es transmitida vía Bluetooth a un Smartphone, donde es posible tener registros diarios de los parámetros descritos.

Incluye un sistema de navegación muy atractivo, que va alojado en el manillar. Las indicaciones son luminosas y de grandes dimensiones, para que no se desvíe demasiado la vista del trayecto, como se muestra en la *figura 1.7*. Incorpora un GPS dentro del cuadro, para localizarla con facilidad. También, relacionado con la seguridad, en la zona trasera integra unas luces de frenado, como es posible observar en la *figura 1.8*.

La bici conecta con un dispositivo móvil para conseguir la información de navegación, o compartir en redes sociales contenidos; todo esto con el objetivo de compartir rutas y experiencias.

Dubike obtiene la energía necesaria para funcionar de un sistema integrado ideado para convertir energía cinética en energía eléctrica, con la que no sólo alimenta a los diferentes sensores, sino que también sirve para encender sus LEDs por la noche y cargar algún teléfono móvil.

Además, tiene un sistema para registrar datos que orienten sobre el estado físico del usuario, informándole de cómo optimizar sus pedaladas o memorizando información sobre distancia recorrida o calorías consumidas.

El proyecto está a cargo de una importante empresa China denominada Bai Du, cuyo objetivo es comercializar la bici a nivel internacional [11].



Figura 1. 7 Vista de manubrio de DUBIKE



Figura 1. 8 Vista trasera de DUBIKE

1.3.4 COBI

COBI es un sistema inteligente para bicicletas cuyo origen se remonta al año del 2014 en Alemania, de la mano de la empresa COBI. Ha sido diseñado para ser utilizado en conjunto con un Smartphone con sistema iOS o Android, estando compuesto por una unidad o soporte delantero con faro incluido, un foco trasero, un mando remoto y una aplicación móvil que controla todo el sistema, tal como se muestra en la figura 1.9.

Su diseño le permite ser instalado en prácticamente cualquier tipo de bicicleta, es un sistema que incluye foco delantero con batería integrada y soporte para Smartphone, foco trasero con luz de freno adaptativa e intermitentes para indicar los giros, un mando remoto para el manillar y una avanzada aplicación compatible con sistemas iOS y Android.

La unidad frontal cuenta con una batería integrada de 6.000 mA que permite recargar algún Smartphone durante el recorrido, siendo a su vez compatible con dinamos para recargar dicha batería con el movimiento producido por el usuario. La unidad cuenta con sensor de luz ambiental, barómetro y acelerómetro.

COBI también integra un pequeño foco trasero que hace las veces de luz de seguridad y luz de freno adaptativa, mostrado en la variando su intensidad según la velocidad a la que se frene.

El sistema es controlado por un mando remoto instalado en el manillar de la bicicleta que nos permite seleccionar las numerosas funciones de COBI mediante cinco sencillos botones, ofreciendo la posibilidad de controlar las luces de la bicicleta, utilizar un navegador GPS (con indicaciones luminosas), realizar el seguimiento de la actividad deportiva, escuchar la música disponible en el dispositivo e incluso de activar una alarma antirrobo que enviará al teléfono la posición GPS de la bicicleta si ésta resulta extraviada [12].

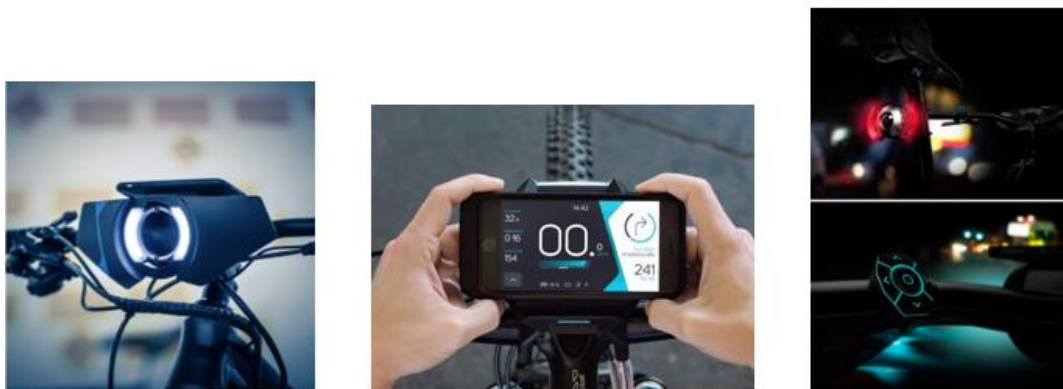


Figura 1. 9 Módulos del sistema para bicicleta COBI

1.3.5 SMART HALO

SmartHalo nace del deseo de un grupo de ciclistas y profesionistas residentes en Canadá que deciden formalizar el proyecto en el año del 2015, estableciendo su empresa oficialmente en la ciudad de Montreal.

Consiste en un gadget que se conecta de forma permanente al manillar de una bicicleta para convertirla en un vehículo inteligente que muestra a cada paso la dirección que se debe tomar, además de las notificaciones del teléfono móvil del usuario a través de luces led intuitivas, como se muestra en la *figura 1.10*.

Por otro lado, también permite saber el lugar donde se ha estacionado la bicicleta gracias a su GPS, tiene luces automáticas para iluminación nocturna, monitoriza el ejercicio realizado y es capaz de gestionar las metas que se le indiquen. Sin embargo, su función más interesante es la de antirrobo, puesto que es capaz de detectar cuando una persona distinta de su dueño se sube a ella y hacer sonar la alarma para disuadir al posible ladrón [13].

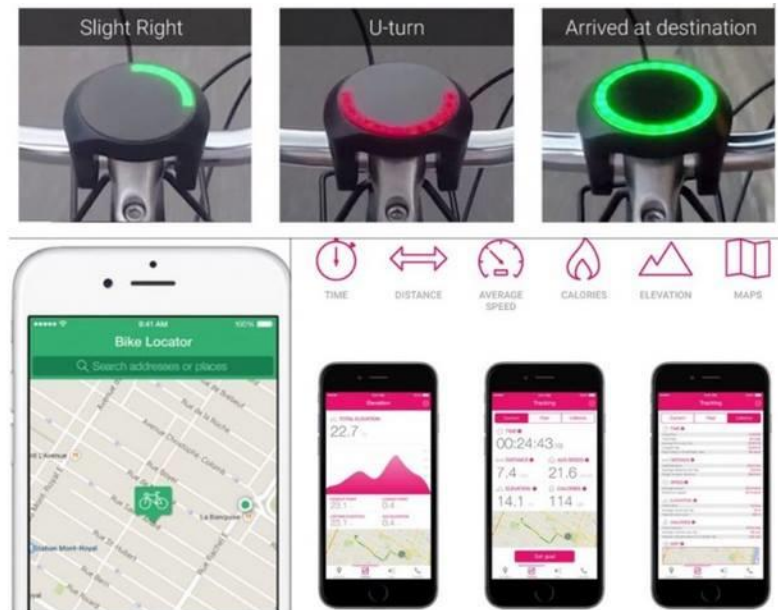


Figura 1. 10 Gadget SmartHalo y su app correspondiente

1.4 LÍNEA DEL TIEMPO DE SISTEMAS PARA BICICLETAS

Con base a la información anteriormente citada, se muestra a continuación en la figura 1.11, la cronología de los proyectos más trascendentes en el ámbito de la tecnología enfocada al ciclismo.

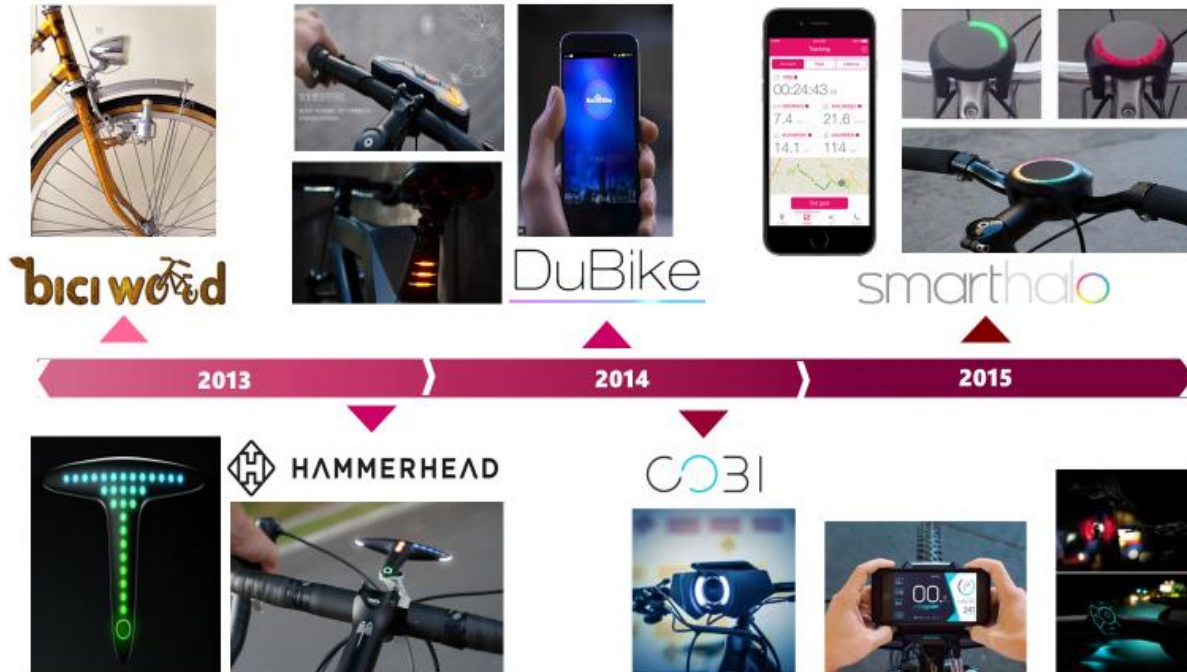


Figura 1. 11 Línea del tiempo de proyectos con sistemas para bicicletas

En general, los sistemas están destinados a satisfacer el mismo conjunto de necesidades a los usuarios: Seguridad vial, sustentabilidad con el ambiente, geolocalización y rendimiento físico, tal como lo muestra la tabla comparativa (Tabla 1.1) en donde es posible apreciar los puntos de convergencia entre los proyectos.

Tabla 1. 1 Funciones básicas de proyectos con sistemas para bicicleta

	BICIWOOD	HAMMERHEAD	DUBIKE	COBI	SMART HALO
APP	X	✓	✓	✓	✓
LUCES	X	✓	✓	✓	✓
GEOLOCALIZACIÓN	X	✓	✓	✓	✓
RENDIMIENTO FÍSICO	✓	X	X	✓	X
RECARGA VÍA USB	✓	X	X	✓	X



CAPÍTULO 2



MARCO TEÓRICO



2

MARCO TEÓRICO

En el presente capítulo se abordan las principales características técnicas de cada uno de los elementos que integran un sistema para bicicleta, así como sus variaciones de uso.

De la misma forma, se muestran las herramientas de diseño y programación necesarias para el desarrollo de una aplicación móvil compatible con el sistema operativo Android.

Por último, se hace un análisis de ciertos procedimientos enfocados en medir la capacidad de las personas en el ámbito del deporte.

2.1 SMARTPHONE

El primer punto que considerar es el hecho de que, debido a los objetivos y alcances del proyecto, el Smartphone empleado dentro del proyecto debe contar con el sistema operativo Android. La mayoría de los dispositivos con Android han incorporado sensores que miden el movimiento, la orientación y otras condiciones ambientales.

Estos sensores son capaces de proporcionar datos en bruto con una gran precisión y exactitud, y son útiles si desea supervisar el movimiento del dispositivo en tres dimensiones o localización, o si desea monitorear los cambios en el entorno ambiental cerca de un dispositivo.

La plataforma Android es compatible con tres amplias categorías de sensores:

- ❖ Sensores de movimiento: Miden las fuerzas de aceleración y las fuerzas de rotación a lo largo de tres ejes. Esta categoría incluye los acelerómetros, sensores de gravedad, giroscopios y sensores de rotación del vector.
- ❖ Sensores ambientales: Miden diferentes parámetros ambientales, como la temperatura del aire ambiente ya la presión, la iluminación, y la humedad. Esta categoría incluye barómetros, fotómetros, y termómetros.
- ❖ Sensores de posición: Miden la posición física de un dispositivo. Esta categoría incluye sensores de orientación y magnetómetros.

El marco sensor de Android le permite acceder a muchos tipos de sensores. Algunos de estos sensores están basados en hardware y algunos son basados en software. La tabla 2.1 resume las principales características de los sensores que son compatibles con la plataforma Android [14].

Tabla 2. 1 Tipos de sensores soportados por la plataforma Android

Sensor	Tipo	Descripción	Usos comunes
ACELERÓMETRO	Hardware	Mide la fuerza de aceleración en m/s^2 que se aplica a un dispositivo en los tres ejes físicos (x, y, z)	Detección de movimiento
TEMPERATURA AMBIENTE	Hardware	Mide la temperatura ambiente de la habitación en grados Celsius ($^{\circ}C$).	Monitoreo de la temperatura del aire.
GRAVEDAD	Software o hardware	Mide la fuerza de la gravedad en m/s^2 que se aplica a un dispositivo en los tres ejes físicos (x, y, z), incluyendo la gravedad	La detección de movimiento
GIROSCOPIO	Hardware	Las medidas de tasa de rotación en m/s^2 alrededor de cada uno de los tres ejes físicos (x, y, z) de un dispositivo.	Detección de rotación
ORIENTACIÓN	Software	Medidas de grados de rotación que hace que un dispositivo de alrededor de los tres ejes físicos (x, y, z).	La determinación de posición del dispositivo.
PRESIÓN	Hardware	Mide la presión del aire ambiente en <u>hPa</u> o mbar.	Seguimiento de los cambios de presión del aire.
VECTOR DE ROTACIÓN	Software o hardware	Mide la orientación de un dispositivo, proporcionando los tres elementos del vector de rotación del dispositivo.	Detección de movimiento y detección de rotación.

2.2 APPS

El desarrollo de las aplicaciones tiene en cuenta las limitaciones de los dispositivos, como la batería o el software; y comienzan probándose, utilizando un emulador, para después ponerse al mercado en versión de prueba [15].

A nivel de programación, existen varias formas de desarrollar una aplicación. Cada una de ellas tiene diferentes características y limitaciones, especialmente desde el punto de vista técnico. El tipo de aplicación que se elija condicionará el diseño visual y la interacción con el usuario.

2.2.1 Aplicaciones Nativas

Las aplicaciones nativas son aquellas que han sido desarrolladas con el software que ofrece cada sistema operativo a los programadores, llamado genéricamente Software Development Kit o SDK [16].

Las aplicaciones nativas se actualizan frecuentemente y en esos casos, el usuario debe volver a descargarlas para obtener la última versión, que a veces corrige errores o añade mejoras.

Una característica poco conocida de las apps nativas es que pueden hacer uso de las notificaciones del sistema operativo para mostrar avisos importantes al usuario, aun cuando no se esté usando la aplicación, un ejemplo son los mensajes de Whatsapp como se muestra en la *figura 2.1*



Figura 2.1 Aplicación nativa en dispositivo Nokia

2.2.2 Aplicaciones Web

La base de programación de las aplicaciones web es el lenguaje de programación HTML, juntamente con JavaScript y CSS, herramientas ya conocidas para los programadores web.

En este caso no se emplea un SDK, lo cual permite programar de forma independiente al sistema operativo en el cual se usará la aplicación. Por eso, estas aplicaciones pueden ser fácilmente utilizadas en diferentes plataformas sin mayores inconvenientes y sin necesidad de desarrollar un código diferente para cada caso particular.

Las aplicaciones web no necesitan instalarse, ya que se visualizan usando el navegador del teléfono como un sitio web normal. Un ejemplo de este tipo de aplicaciones es Facebook que cuenta con una web app, como se puede observar en la *figura 2.2*.



Figura 2. 2 Aplicación web en dispositivo Apple

2.2.3 Aplicaciones Híbridas

Este tipo de aplicaciones es una especie de combinación entre las dos anteriores. La forma de desarrollarlas es parecida a la de una aplicación web es usando HTML, CSS y JavaScript, y una vez que la aplicación está terminada, se compila o empaqueta de forma tal, que el resultado final es como si se tratara de una aplicación nativa.

Esto permite casi con un mismo código obtener diferentes aplicaciones, por ejemplo, para Android y iOS, y distribuirlas en cada una de sus tiendas.

A diferencia de las aplicaciones web, estas permiten acceder, usando librerías, a las capacidades del teléfono, tal como lo haría una app nativa. Un ejemplo de este tipo de apps es Netflix que tiene una aplicación híbrida que se ve prácticamente igual en iOS y en Android, como se muestra en la *figura 2.3*.



Figura 2. 3 Aplicación híbrida en dispositivos con sistema operativo IOS y Android

2.3 SOFTWARE DE DISEÑO

App Inventor es un entorno de desarrollo de software, originalmente desarrollado por Google Labs y ahora mantenida por el Instituto de Tecnología de Massachusetts (MIT), para la creación de aplicaciones compatibles con el sistema operativo Android.

Utiliza una interfaz gráfica que permite a los usuarios, de forma visual y a partir de un conjunto de herramientas básicas, arrastrar y soltar objetos que constituyen el diseño de la propia app.

Así mismo, permite ir enlazando una serie de bloques con funciones específicas, lo que constituye la lógica de programación de la aplicación.

Además, App Inventor se basa en un servicio web que permite almacenar los trabajos realizados y ayuda a dar un seguimiento a los proyectos.

App Inventor incluye los siguientes componentes [17]:

- ❖ Un diseñador, en el que se especifican los componentes de un programa. Esto incluye componentes visibles, como botones e imágenes, que se colocan en una pantalla simulada, y los componentes no visibles, tales como sensores y conexiones web.

- ❖ Un editor de bloques, en el que se crea la lógica del programa.
- ❖ Un compilador basado en el lenguaje Kawa.
- ❖ Una aplicación para la depuración en tiempo real sobre un dispositivo Android.

App Inventor permite también ejecutar las aplicaciones en un emulador, por lo que no es imprescindible disponer del teléfono para probar los programas que se hacen.

El procedimiento de desarrollo de una aplicación haciendo uso de la plataforma App Inventor queda ilustrado en la figura 2.4. Es posible apreciar que el proyecto es creado y almacenado en el servidor de Google, desde donde se pueden hacer modificaciones tanto de diseño como de funcionalidad. Los avances del trabajo pueden ser visualizados a través del emulador contenido en la plataforma o directamente en el dispositivo móvil, descargando e instalando el archivo apk, dando lugar a una nueva app.

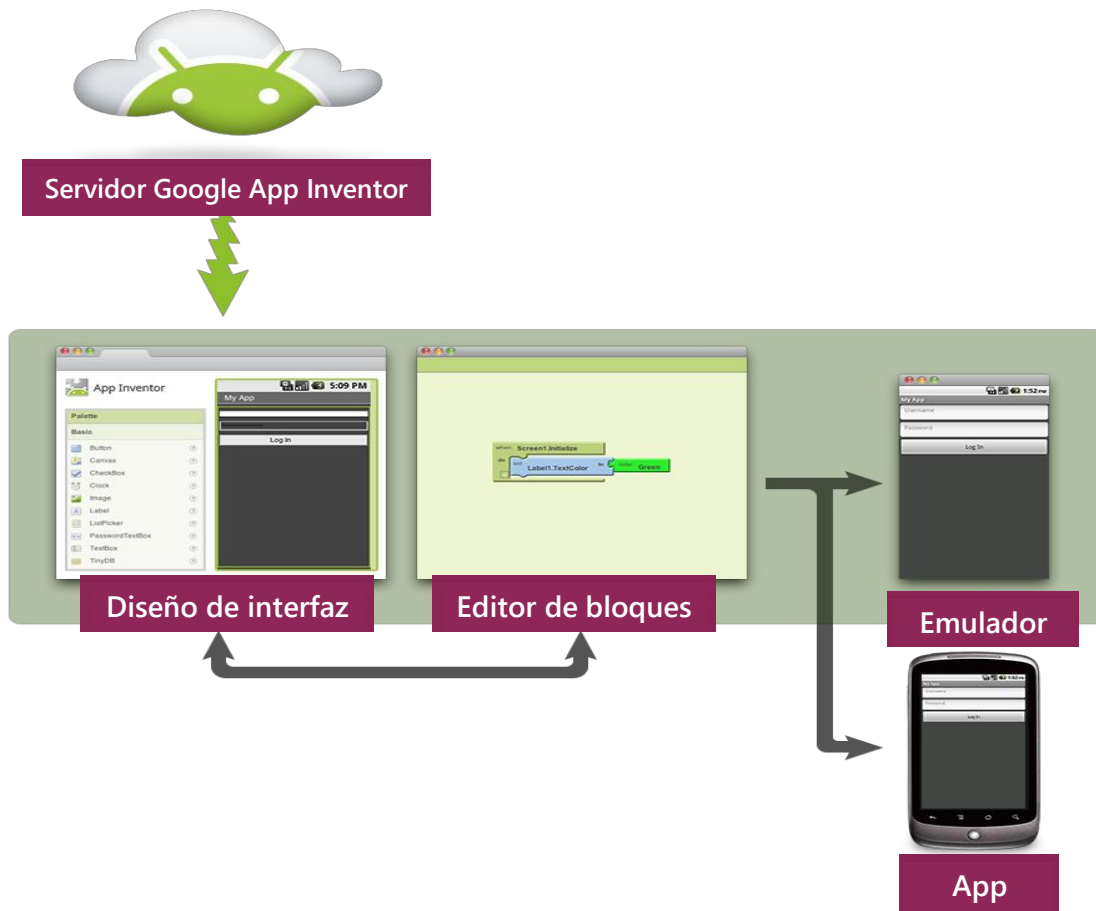
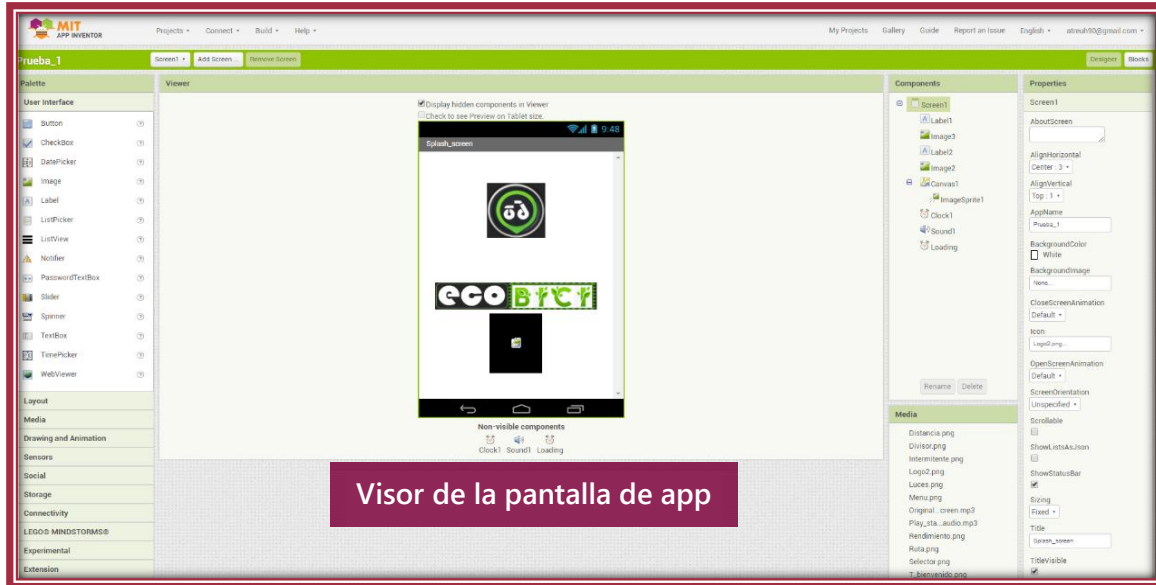


Figura 2. 4 Desarrollo de una App a través de la plataforma MIT App Inventor 2

En las figuras 2.5 y 2.6 se muestra la plataforma de trabajo ofrecida por App Inventor a los usuarios para la creación de aplicaciones móviles.



Paleta	Medios	Propiedades
Interfaz de usuario <ul style="list-style-type: none"> Botón CasillaDeVerificación SelectorDeFecha Imagen Etiqueta SelectorDeLista VisorDeLista Notificador CampoDeContraseña Deslizador Desplegable CampoDeTexto 	<ul style="list-style-type: none"> Distancia.png Divisor.png Intermitente.png Logo2.png Luces.png 	Screen1 <p>PantallaAcercaDe</p> <p>AccentColor</p> <p>Por defecto</p> <p>DispHorizontal</p> <p>Centro : 3</p> <p>DispVertical</p> <p>Arriba : 1</p> <p>AppName</p> <p>Eco_Bici</p> <p>ColorDeFondo</p> <p>Por defecto</p> <p>ImagenDeFondo</p> <p>Ninguno...</p> <p>AnimaciónCierreDePantalla</p> <p>Por defecto</p>
	Componentes <ul style="list-style-type: none"> Screen1 <ul style="list-style-type: none"> Label1 Image3 Label2 Image2 Canvas1 <ul style="list-style-type: none"> ImageSprite1 Clock1 	

Figura 2. 5 Plataforma de diseño de MIT App Inventor 2

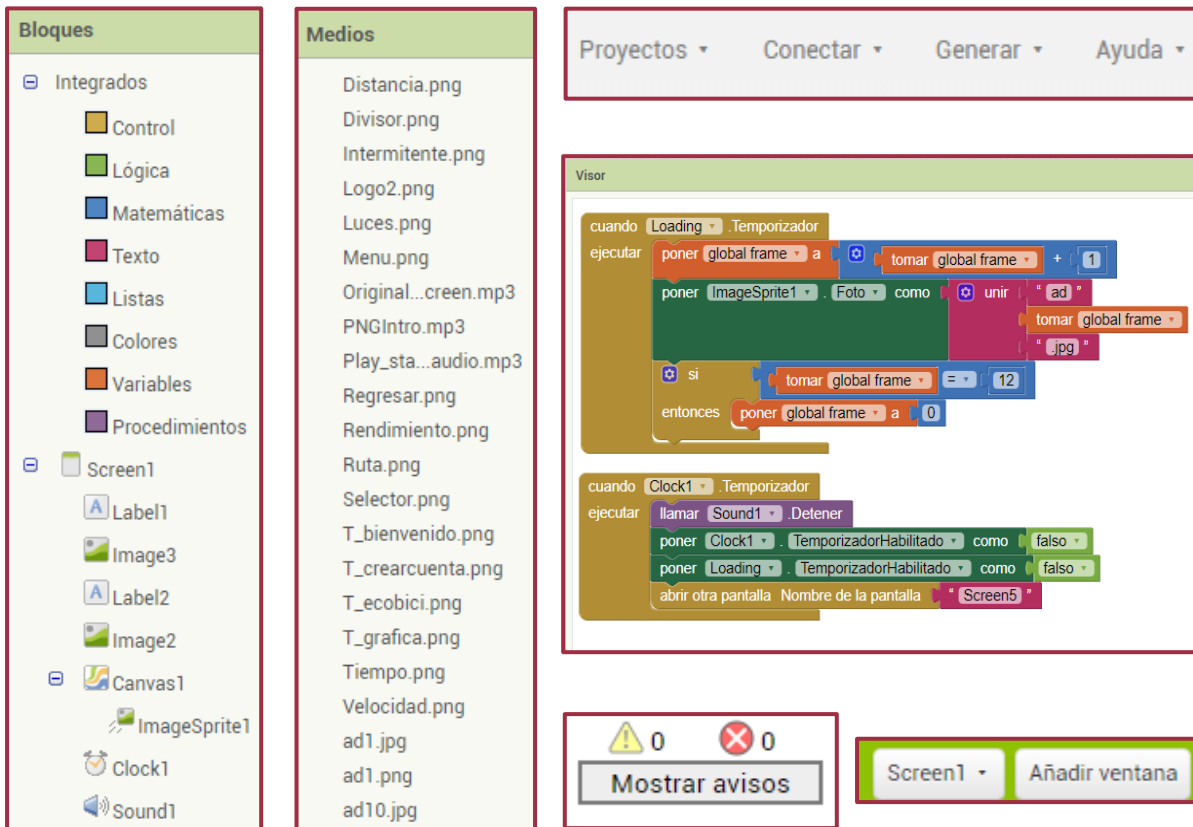
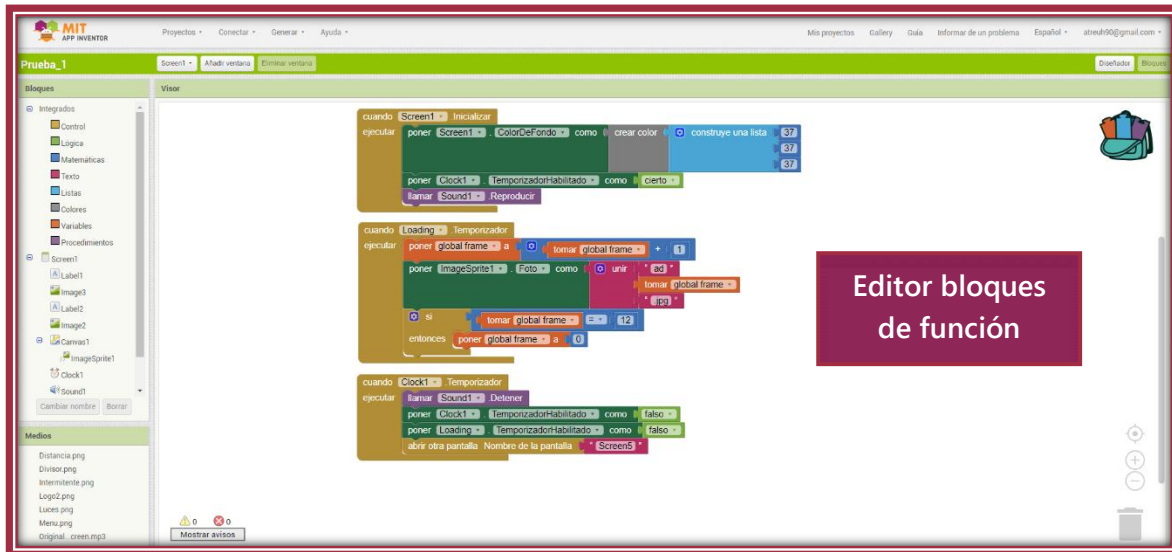


Figura 2. 6 Editor de bloques de MIT App Inventor 2

2.4 ARDUINO

Arduino es una plataforma electrónica de código abierto basada en hardware y software flexibles y fáciles de usar. El microcontrolador de la placa se programa usando el Lenguaje de Programación de Arduino y el Entorno de Desarrollo de Arduino [18].

Arduino simplifica el proceso de trabajo con microcontroladores, ofreciendo también otras notables ventajas sobre los demás sistemas:

- ❖ Multiplataforma: El software de Arduino se ejecuta en sistemas operativos Windows, Macintosh OSX y GNU/Linux.
- ❖ Código abierto y software extensible: El software Arduino está publicado como herramientas de código abierto, disponible para extensión por programadores experimentados.
- ❖ Código abierto y hardware extensible: El Arduino está basado en microcontroladores ATMEGA8 y ATMEGA168 de Atmel. Los planos para los módulos están publicados bajo licencia, por lo que diseñadores experimentados de circuitos pueden hacer su propia versión del módulo, extendiéndolo y mejorándolo.

2.4.1 Arduino UNO

El Arduino UNO contiene todos los componentes elementales para soportar aplicaciones no muy complejas, su versatilidad y facilidad de uso lo hacen ideal para proyectos que no requieren gran cantidad de recursos.

Cuenta con un microcontrolador basado en el Atmega328P. Tiene 14 pines que funcionan como entradas y salidas, de las cuales 6 pueden utilizarse como salidas PWM, y 6 entradas analógicas, un cristal de cuarzo a 16 MHz y una conexión USB, su alimentación es de 5V y su sistema es de 8 bits, tal y como se muestra en la Figura 2.7 [19].



Figura 2. 7 Arduino UNO

2.5 MÓDULO BLUETOOTH

Bluetooth es una especificación industrial para Redes Inalámbricas de Área Personal (WPANs) que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia en la banda ISM de los 2,4 GHz [20].

Los principales objetivos que se pretenden conseguir con esta norma son:

- ❑ Facilitar las comunicaciones entre equipos móviles y fijos.
- ❑ Eliminar cables y conectores entre éstos.
- ❑ Ofrecer la posibilidad de crear pequeñas redes inalámbricas y facilitar la sincronización de datos entre equipos personales.

El módulo Bluetooth HC-06 y el HC-05, gracias a sus puertos Tx (Transmisión) y Rx (Recepción) nos permiten realizar comunicaciones inalámbricamente a 10 metros de distancia como máximo, resultando muy fáciles de manejar gracias a su SPP (Serial Port Protocol) de bajo consumo, ya que trabajan a 3.3V.

El módulo de Bluetooth HC-06 solo opera de modo esclavo, a diferencia de su similar HC-05, que ofrece una mejora, con respecto a precio y características, ya que es un módulo Maestro-Esclavo, esto quiere decir que además de recibir conexiones desde una PC o Tablet, también es capaz de generar conexiones hacia otros dispositivos Bluetooth.

El módulo Bluetooth tiene unas medidas extremadamente pequeñas de 26.9x13mm y un grosor de 2.2mm, es idóneo para utilizar en cualquier aplicación sin ocupar mucho espacio tal y como se muestra en la *figura 2.8*.

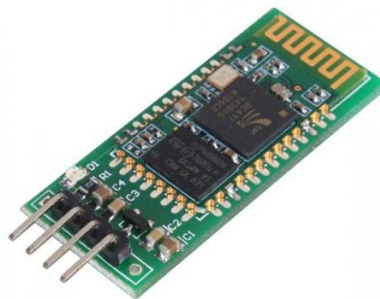


Figura 2. 8 Módulo Bluetooth HC-05

2.6 RELEVADOR

En los circuitos electrónicos, a veces es necesario controlar una carga de potencia a partir de una señal digital de 5v. Una forma de conseguirlo, aislando la zona de 5v y la de alta tensión, puede ser mediante la utilización de un relé, ilustrado en la *figura 2.9*.

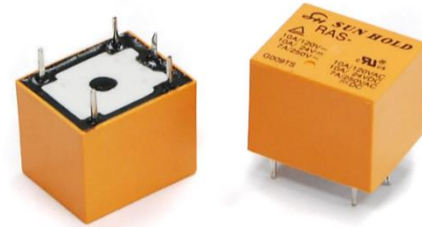


Figura 2. 9 Relevador electromagnético

El relé o relevador es un dispositivo electromecánico. Funciona como un interruptor controlado por un circuito eléctrico en el que, por medio de una bobina y un electroimán, se acciona un juego de uno o varios contactos que permiten abrir o cerrar otros circuitos eléctricos independientes.

Dado que el relé es capaz de controlar un circuito de salida de mayor potencia que el de entrada, puede considerarse, en un amplio sentido, como un amplificador eléctrico.

2.7 LUCES

Al igual que existe una clasificación dentro de las bicicletas, se tiene una clasificación para las luces que se pueden utilizar teniendo en cuenta que existen luces frontales y traseras, es por ello que se opta por implementar en cuanto a luces frontales las luces urbanas debido a su gran versatilidad para su instalación en los diferentes tipos de bicicletas.



Figura 2. 10 Luz urbana rural

Las luces urbanas tienen muchas características que las hacen sobresalir como la calidad de construcción, lo fácil de usar y la robusta montura que tienen, la óptica, los LEDs de lado extras y todos los pequeños detalles suman, ejemplo de éste que se observa en la *figura 2.10* [21].

De igual forma basándonos en el Manual del Ciclista Urbano de la Ciudad de México [22] que recomienda el uso de accesorios como las luces, mencionando, “Como cualquier otro vehículo, la bicicleta debe contar con un sistema básico de luces fijas para circular de noche: una lámpara de luz blanca delantera y una luz roja intermitente trasera”.

Basándonos de nuevo en el Manual del Ciclista, podemos decir en cuanto a la luz trasera que, debe cumplir con dos requerimientos fundamentales: uno de ellos es que sea una luz roja y otra es que sea intermitente, ya que con éste se busca que otros conductores de vehículos motorizados nos vean y también poder realizar señalamientos de las maniobras del ciclista. Un tipo de luz trasera de LEDs se puede ver en la *figura 2.11*.



Para la selección de las luces a implementar se debe tomar en cuenta la norma ANSI FL1 [23].

Figura 2. 11 Luz LED trasera

Generalmente se hace la instalación de la luz frontal en el manubrio de la bicicleta, donde se ésta ilumine la ruta que estamos siguiendo y la luz trasera se ubica debajo del asiento.

2.8 SENSOR DE PULSO CARDIACO

Es un sensor de ritmo cardiaco óptico tipo *plug and play* compatible con la tarjeta Arduino. Contiene una etapa de amplificación y un filtro para el ruido, lo cual hace que su señal de salida sea confiable y estable. El consumo de corriente es bajo siendo de 4mA con una alimentación de 5V. Esta basado en un LED emisor y un sensor receptor de intensidad, la cantidad de luz reflejada por el dedo cuando hay paso de corriente sanguínea define la salida del sensor. Por lo que es posible visualizar gráficamente o numéricamente la información del mismo.

Las especificaciones técnicas son:

Voltaje de Operación: 3.0V – 5.5V DC

Consumo corriente: 20mA máx.

Sensor: APDS-9008

Opamp: MCP6001

Led verde

Longitud de cable: 20cm

Cables: GND, VCC, Señal



Figura 2. 12 Sensor de pulso cardiaco



2.9 ESTIMACIÓN DE RENDIMIENTO FÍSICO

El cuerpo humano es la suma de muchos sistemas y cada uno de estos es, así mismo, la suma de muchos conceptos. Por ende, para valorar su nivel de condición física, lo primero que se debe hacer es trazar variables medibles.

Evidentemente, a día de hoy, se desconocen muchas cosas sobre nuestro organismo y eso significa que no es posible medirlo todo. Sin embargo, una vez que se consigue medir ciertas variables al momento de realizar alguna actividad física, es posible observar y cuantificar la evolución en el desempeño de la persona cada vez que se somete a dicho ejercicio.

Los músculos, para trabajar en óptimas condiciones necesitan las cantidades adecuadas de dos elementos que están relacionados: energía y oxígeno. La energía para mover los músculos se obtiene de una molécula llamada ATP. Este se obtiene de los alimentos, que se van rompiendo en partes muy pequeñas, formando moléculas como glucosa, fructosa y galactosa. La gran mayoría es glucosa y el resto puede pasar a ella. Esta glucosa es la que se rompe formando ATP.

La producción de ATP en cantidades suficientes para apoyar la actividad muscular continua requiere del ingreso de oxígeno. Si no existe suficiente oxígeno, la obtención de energía en forma de ATP, será menor. Por esto es de vital importancia el oxígeno aportado por la respiración. Esta capacidad, determinada por la habilidad del organismo para captar, transportar y utilizar oxígeno durante la actividad física se denomina Consumo Máximo de Oxígeno (VO_2).

El VO_2 es un parámetro que indica la capacidad aeróbica de trabajo por unidad de tiempo y refleja, de forma global, el sistema de transporte de oxígeno desde la atmósfera hasta su utilización en el músculo. Por tanto, es un fiel indicador de la capacidad y eficiencia aeróbica de una persona. A mayor consumo de oxígeno, mayor nivel de condición física cardiovascular [24].

Por otra parte, para optimizar los programas de ejercicio es necesario el control calórico de la actividad, ya que, el conocer cuántas calorías consume la persona en una actividad aeróbica determinada establece un panorama para ajustar la ingesta calórica diaria y potenciar su rendimiento a través de una alimentación adecuada.

Las calorías son la unidad de medida que se usa para saber la cantidad de energía que nos aportan los alimentos. La mayoría de las personas necesitan entre 1,500 y 2,000 calorías al día. Sin embargo, para muchos deportistas, esta cifra puede aumentar de 1,000 a 1,500 calorías más.

La conversión de las calorías de su estado normal (nutrientes) a la forma que puede ser quemada por las células del músculo durante el ejercicio aeróbico se logra,



mediante el proceso de la respiración celular, que requiere la entrega de oxígeno a través del torrente sanguíneo hacia el músculo y está directamente relacionado con su frecuencia cardíaca. Es esta relación la que permite predecir el gasto de energía (es decir, la cantidad de calorías quemadas).

En esencia, con la intensidad del ejercicio los músculos deben quemar más calorías, por lo que el corazón debe latir más rápido para proporcionar el oxígeno necesario para convertir las calorías en energía que pueda ser quemada por los músculos.

2.10 FRECUENCIA CARDIACA

La frecuencia cardíaca (F_c) es uno de los indicadores fisiológicos más importantes para el entrenamiento deportivo, las definiciones de (F_c) hacen referencia al número de latidos por minuto (Lpm) al que se contrae el músculo cardíaco. Está condicionada principalmente por el tipo de ejercicio realizado; aunque también existen otros factores para tener en cuenta como sexo, edad, peso, ambiente, así como también las amenazas a la seguridad y las respuestas emocionales, entre otros.

El corazón tiene un sistema que a través de un impulso eléctrico le dice cuándo latir y bombear la sangre alrededor del cuerpo. Un sistema eléctrico defectuoso puede derivar en un ritmo cardíaco anómalo. Si bien es cierto que una frecuencia cardíaca normal no garantiza que una persona esté libre de problemas de salud, es un punto de referencia útil para identificar una variedad de enfermedades.

Los tipos más importantes de trastornos del ritmo cardíaco incluyen:

Taquicardia: Un ritmo cardíaco rápido (más de 100 latidos por minuto)

Bradycardia: Un ritmo cardíaco lento (menos de 60 latidos por minuto)

Arritmias supraventriculares: Ritmos anormales que se originan en las cámaras superiores del corazón (aurículas)

Arritmias ventriculares: Arritmias que se originan en las cámaras inferiores del corazón (ventrículos)

Bradiarritmias: Ritmos lentos del corazón que generalmente están causados por una enfermedad en el sistema de conducción del corazón

El pulso se puede tomar en cualquier arteria superficial que pueda comprimirse contra un hueso. Los sitios donde se puede tomar el pulso son:

En la sien (temporal).

En el cuello (carotídeo).

Parte interna del brazo (humeral).

En la muñeca (radial).

Parte interna del pliegue del codo (cubital).

2.11 DIAGRAMA A BLOQUES DEL SISTEMA PARA BICICLETA

En la *figura 2.13*, se muestra a grandes rasgos el funcionamiento y la interacción que tienen los elementos que conforman el sistema para bicicleta en este proyecto.

Se infiere que el elemento central del sistema consiste en una app, compatible con el sistema operativo Android y desarrollada a partir de la plataforma de App Inventor 2. La aplicación, recibe información primeramente del usuario el cual, después de haber creado una cuenta, introduce datos personales básicos, fundamentales para estimación de su estado físico actual. Así mismo, la app es capaz de obtener información sobre el recorrido del usuario a partir de la lectura de los estados de los sensores internos del Smartphone, ubicándolo en el tiempo y el espacio.

Una vez procesada dicha información, se muestra por medio de la aplicación, en tiempo real, el progreso de su recorrido basada en la geolocalización del dispositivo y una estimación del rendimiento físico del usuario, incluyendo una gráfica de pulso cardiaco.

Paralelamente, la aplicación está vinculada, a través de un módulo bluetooth, con la plataforma Arduino, lo que hace posible para el usuario poder controlar la iluminación frontal y posterior de la bicicleta por medio de un módulo de relevadores y a su vez recibir información por parte de un sensor de pulso cardiaco colocado directamente sobre el cuerpo del usuario.

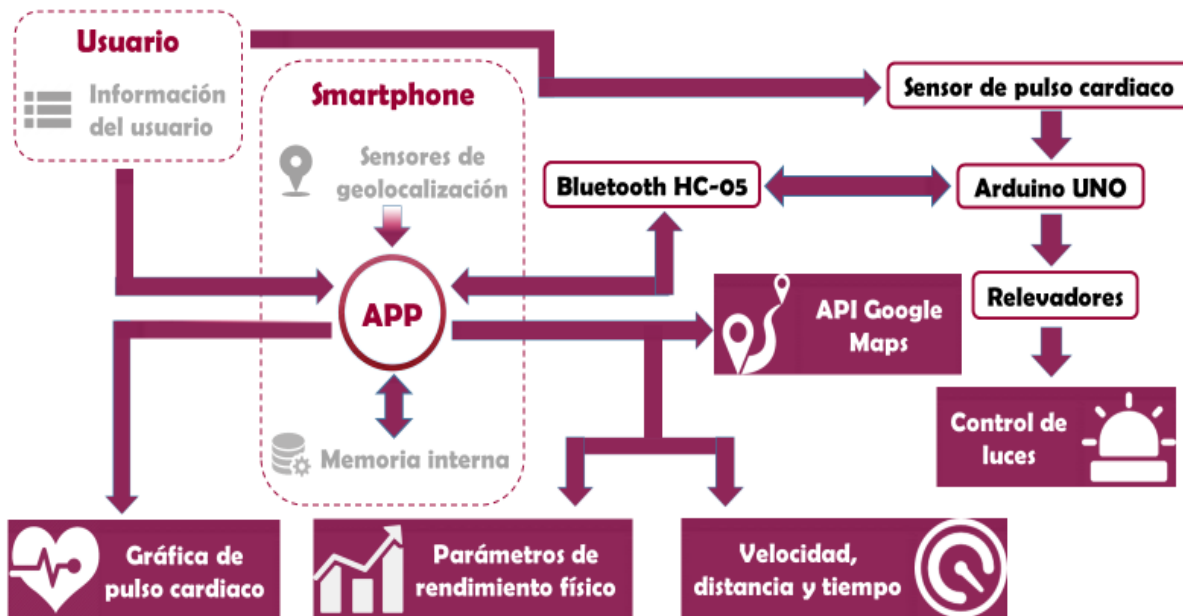


Figura 2.13 Diagrama a bloques del sistema para bicicleta



CAPÍTULO 3



DISEÑO E IMPLEMENTACIÓN



3

DISEÑO E IMPLEMENTACIÓN

En el presente capítulo se explica propiamente el proceso de desarrollo del sistema:

- ❖ Los recursos de software empleados en el diseño y programación de las diferentes interfaces que conforman la app, desde los bocetos hasta su versión final.
- ❖ El establecimiento de la transferencia de información entre el dispositivo móvil (Smartphone) y los diversos módulos y dispositivos que conforman el hardware del sistema.
- ❖ Los algoritmos y modelos matemáticos empleados para la obtención de los parámetros de rendimiento físico del usuario de la app.

3.1 DESARROLLO DE APP

Dentro del proceso de desarrollo de la app es necesario considerar, una vez teniendo claramente establecidos los objetivos del proyecto, el diseño de la misma. En este punto, la creación de los bocetos que dan origen a la aplicación se realiza mediante el programa de presentación Power Point, de Microsoft Office, debido a la gran variedad de herramientas de diseño que ofrece al usuario.

Una vez obtenidos los bocetos, es necesario hacer uso de un kit de desarrollo de software (SDK) que permita plasmar las ideas vertidas en los bocetos y darles funcionalidad a través del desarrollo de un programa, específicamente una app nativa, que sea compatible con el sistema operativo Android y que permita hacer uso de los recursos integrados al Smartphone a emplear.

Entre la diversidad de software con esas características, MIT App Inventor 2 es el que representa la opción más viable tomando en consideración la amplia gama de necesidades y recursos que abarca en cuestión de aplicaciones móviles, a través de una plataforma visualmente atractiva y relativamente fácil de manipular.

Por otra parte, al tratarse de una plataforma Web no es necesaria su instalación dentro de un dispositivo o computadora, sino que la creación y progresos del proyecto en cuestión se almacenan en la cuenta del usuario, dentro de la página web ai2.appinventor.mit.edu, como se muestra en la *figura 3.1*.

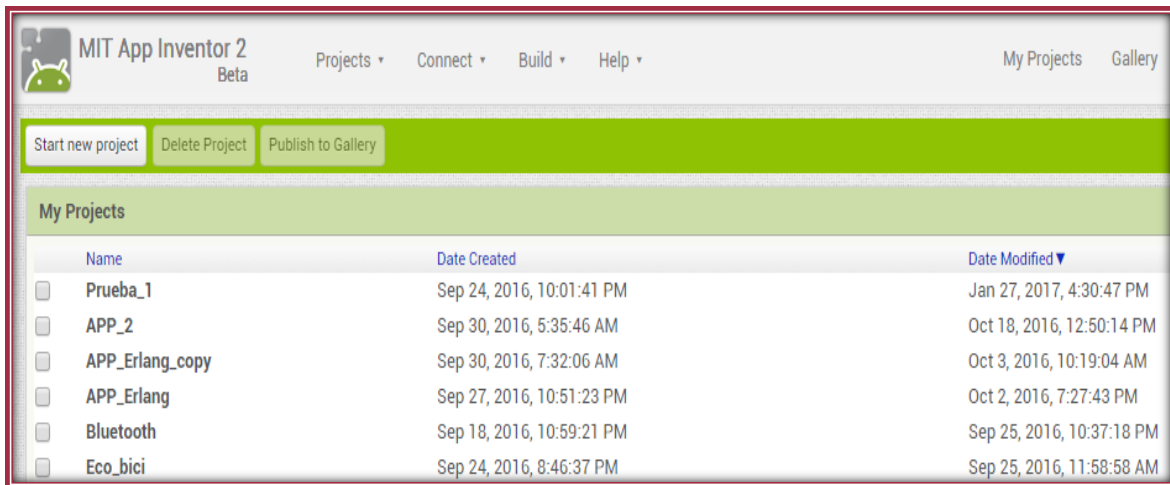


Figura 3. 1 Lista de proyectos creados en una cuenta de MIT App Inventor 2

Habiendo creado un proyecto, es posible agregarle funciones o añadirle nuevos elementos a través de dos interfaces. La primera de ellas enfocada en la parte de diseño, es decir la interfaz con la cual interactúa el usuario de la app. En ella es posible establecer la distribución de los elementos a lo largo de la pantalla, agregar componentes o archivos tales como: texto, imágenes, audio, botones, etc., así como modificar sus propiedades, e incluso, hacer uso de recursos del sistema en donde se instalará la app como, por ejemplo: sensores de geolocalización, cronómetros, componentes bluetooth, entre otros, como se aprecia en la *figura 3.2*.

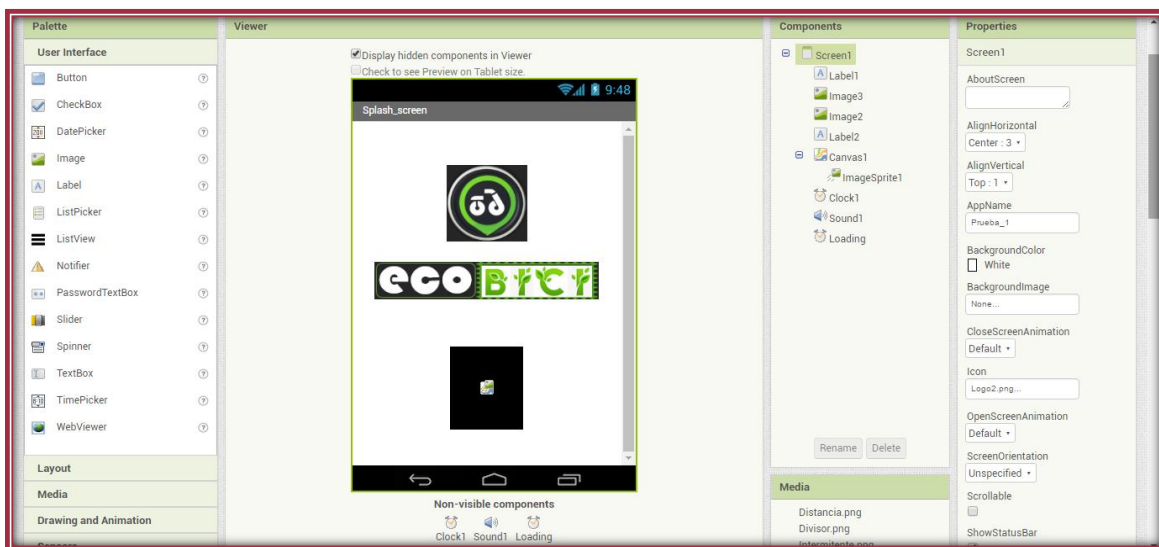


Figura 3. 2 Plataforma de diseño de app en MIT App Inventor 2

La otra sección, se enfoca en desarrollar las sentencias lógicas del programa a través de bloques que, de acuerdo a las funciones desempeñadas dentro de un código, son clasificados en diferentes áreas (Lógicos, de texto, matemáticos, de control, variables, etc.) e identificados a través de diferentes colores y contienen ranuras específicas para ser unidos únicamente con aquellos bloques cuyas ranuras puedan embonarse, esto con la finalidad de evitar fallas de sintaxis dentro del programa, como se puede percibir en la *figura 3.3*.

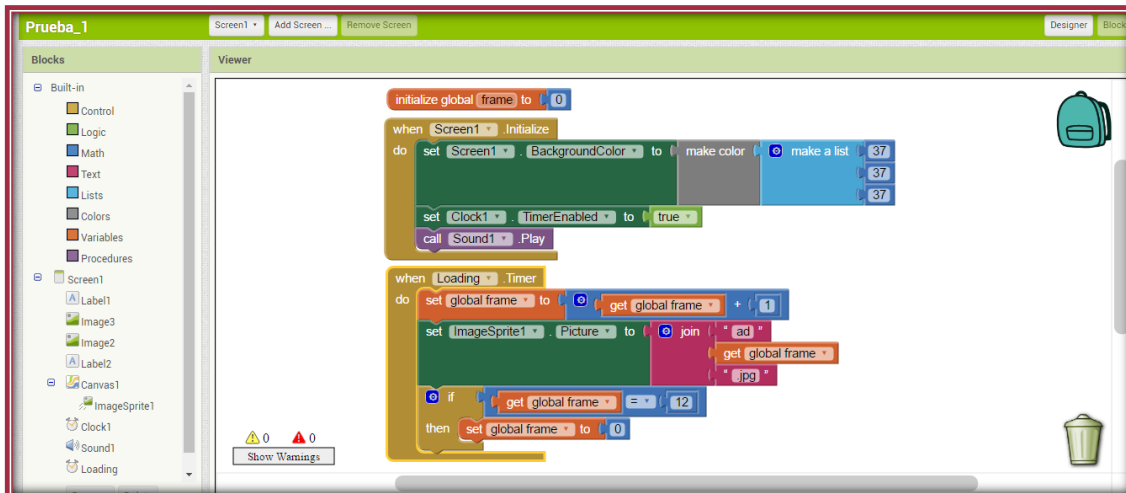


Figura 3. 3 Plataforma de bloques de app en MIT App Inventor 2

Cada vez que se requiere probar el funcionamiento de cierto proyecto creado, MIT App Inventor ofrece la opción de instalar y correr la app directamente en el Smartphone a través de la transferencia de un archivo con el formato APK desde la computadora hacia el Smartphone.

La segunda opción consiste en hacer uso del emulador que contiene la plataforma de MIT App Inventor 2, desde el cual se puede verificar el progreso de las apps sin necesidad del dispositivo físico.

Es importante mencionar que, debido a la multiplicidad de funciones para las cuales está destinada la app, es, necesaria la creación de diversas interfaces, cada una con elementos y componentes, tanto de programación como de diseño, distintos y muy particulares, que se enfocan en realizar tareas específicas, pero dentro de un mismo proyecto.

La evolución en el desarrollo de cada una de las pantallas que constituyen la app se muestra a continuación.



```

when Screen1 Initialize
do
  set Screen1 BackgroundColor to make color 0 make a list 37
  set Clock1 TimerEnabled to true
  call Sound1 Play
  
```

Inicializar con color de fondo, activar el cronometro y el sonido

```

when Loading Timer
do
  set global frame to get global frame + 1
  set ImageSprite1 Picture to join ad get global frame jpg
  if get global frame = 12
  then set global frame to 0
  
```

Animación de la imagen de carga (Splash screen)

```

when Clock1 Timer
do
  call Sound1 Stop
  set Clock1 TimerEnabled to false
  set Loading TimerEnabled to false
  open another screen screenName Screen5
  
```

Abrir pantalla de validación de usuario

(d) Componentes de bloques App Inventor



Figura 3. 4 Proceso de elaboración del Splash Screen o pantalla de inicio



```

if [get global búsqueda] = [select list item list index]
then
  set global indice to [get global index]
  set global index to [get global index] + 1

```

Validación de usuario

```

if [get global password] = [select list item list index]
then
  call Bd_persona StoreValue
  tag [Indice]
  valueToStore [get global indice]
  open another screen screenName [Screen2]
else
  set Error_contraseña Visible to true

```

Validación de contraseña

```

when [BT_Cuenta] Click
do
  open another screen screenName [Screen8]

```

Abrir pantalla para crear una nueva cuenta

(d) Componentes de bloques App Inventor



Figura 3. 5 Proceso de elaboración de la pantalla de validación de usuario



```

if PT_contraseña2 . Text ≠ PT_contraseña2 . Text
then set advertencia . Visible to true
else
  add items to list list get global nombre
  item TB_nombre . Text
  add items to list list get global email
  item TB_email . Text
  add items to list list get global usuario
  item TB_usuario . Text
  add items to list list get global contraseña
  item PT_contraseña . Text
  add items to list list get global edad
  item TB_edad . Text
  add items to list list get global peso
  item TB_peso . Text
  add items to list list get global estatura
  item TB_estatura . Text

```

Guardar datos del nuevo usuario en una base de datos

```

if Mujer . Checked = true
then
  add items to list list get global vo2
  item select list item list get global vo_mujer
  index get global indice2
  set global frec_max2 to 208.1 - 0.77 * get global edad2
  add items to list list get global frec_max
  item get global frec_max2
  add items to list list get global sexo
  item Mujer

```

Definir frec. max2 y vo2 de acuerdo con el sexo y edad

(d) Componentes de bloques App Inventor



Figura 3. 6 Proceso de elaboración de interfaz para dar de alta una cuenta



(a) Vista previa App Inventor

(b) Base de datos

```

when Screen1.Initialize
do
  set Screen1.BackgroundColor to make color 233 190 222
  set global nombre to call TinyDB1.GetValue
  tag Nombre
  valueIfTagNotThere create empty list
  set global sexo to call TinyDB1.GetValue
  tag Sexo
  valueIfTagNotThere create empty list
  set global email to call TinyDB1.GetValue
  tag Email
  valueIfTagNotThere create empty list
  set global usuario to call TinyDB1.GetValue
  tag Usuario
  valueIfTagNotThere create empty list
  set global contraseña to call TinyDB1.GetValue
  tag Contraseña
  valueIfTagNotThere create empty list
  set global edad to call TinyDB1.GetValue
  tag Edad
  valueIfTagNotThere create empty list
  set global peso to call TinyDB1.GetValue
  tag Peso
  valueIfTagNotThere create empty list
  set global estatura to call TinyDB1.GetValue
  tag Estatura
  valueIfTagNotThere create empty list
  set ListUsuario.Elements to get global usuario
  set ListSexo.Elements to get global sexo
  set ListEdad.Elements to get global edad
  set ListPeso.Elements to get global peso
  set ListEstatura.Elements to get global estatura
  
```

Mostrar el contenido de las listas que conforman la base de datos

(c) Componentes de bloques App Inventor



(d) Componentes de diseño App Inventor

Figura 3. 7 Proceso de elaboración de la base de datos



```

when LocationSensor1 LocationChanged
latitude longitude altitude speed
do
set global contador to get global contador + 1
set global velocidad to get speed
set global UbicacionesRuta to 
set TB_velocidad Text to join get global velocidad
km / h
add items to list list get global LatitudesRuta
item get latitude
add items to list list get global LongitudesRuta
item get longitude
call Trazado_Ruta
call Frecuencia_Calorias
call Maximos_Totales
call FormatoDistancia
set global cambio_pantalla to 0
call Guardar_variables
  
```

Velocidad, distancia y geolocalización en tiempo real

```

when Tiempo_recorrido Timer
do
set global segundos to get global segundos + 1
if get global segundos = 60
then
set global segundos to 0
set global minutos to get global minutos + 1
if get global minutos = 60
then
set global minutos to 0
set global horas to get global horas + 1
  
```

Creación del cronometro del recorrido

(d) Componentes de bloques App Inventor



Figura 3. 8 Proceso de elaboración del mapa de geolocalización y medidor de velocidad



(a) Vista previa App Inventor

(c) Resumen de ruta

```

when Screen1 Initialize
do
  set Screen1 BackgroundColor to make color 233 190 22
  call Inicializar_variables
  if get global bandera_consulta = 1
  then
    set global url to join file://mnt/sdcard/Mimapa
    get global indice
    get global indice_lista + 1
    html
    call WebViewer1 GoToUrl
    url get global url
    set Lb Fecha Text to join get global fecha_hora
    get global fecha_hora2
    set Lb tiempo Text to get global tiempo
    set Lb distancia Text to get global distancia
    set Lb velocidadmedia Text to join round get global velocidad_media
    km/h
    set Lb velocidadmax Text to join round get global velocidad_max
    km/h
    set Lb frecuenciaimedia Text to join round get global frecuencia_media
    ppm
    set Lb frecuenciaimax Text to join round get global frecuencia_max
    ppm
    set Lb calorias Text to join round get global calorias
    Kcal
    set Lb cargaaerobica Text to join round get global carga_aerobica
    %
  
```

Mostrar resumen de geolocalización y rendimiento físico

(d) Componentes de bloques App Inventor



(e) Componentes de diseño App Inventor

Figura 3. 9 Proceso de elaboración del resumen del recorrido y mapa de ruta



(a) Vista previa App Inventor

(c) Historial de rutas

```

if [get global indice_lista] >= [get global acumulador]
then
  set Ruta10 Visible to true
  set Lb_ruta10 Text to [join [RUTA] [get global acumulador]]
  set Lb_fecha10 Text to [join [select list item list [get global lista_fecha] [index [get global acumulador]]] [select list item list [get global lista_horario2] [index [get global acumulador]]]]
  set global acumulador to [get global acumulador] + 1

```

Mostrar historial de rutas del usuario con fecha y hora

```

when Bt flecha1 Click
do
  set global numero_recorrido to 1
  set global bandera_consulta to 1
  call TinyDB1 StoreValue
  tag Numero_recorrido
  valueToStore [get global numero_recorrido]
  call TinyDB1 StoreValue
  tag Bandera_consulta
  valueToStore [get global bandera_consulta]
  open another screen screenName Screen11

```

Llevar a pantalla de resumen de la ruta seleccionada

(d) Componentes de bloques App Inventor



(e) Componentes de diseño App Inventor

Figura 3. 10 Proceso de elaboración de historial de recorridos



```
when ListPicker1 . BeforePicking
do set ListPicker1 . Elements to BluetoothClient1 . AddressesAndNames
```

Mostrar lista de los dispositivos vinculados con el smartphone a través de Bluetooth

```
when ListPicker1 . AfterPicking
do set ListPicker1 . Selection to call BluetoothClient1 . Connect
address ListPicker1 . Selection
```

Establecer conexión Bluetooth con el dispositivo seleccionado

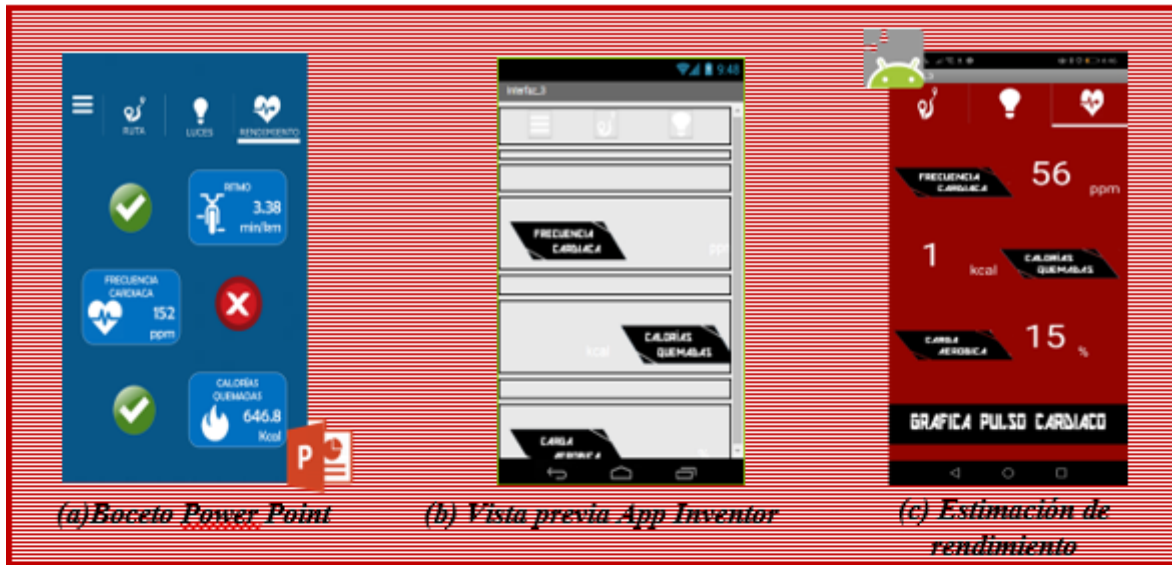
```
when BT_encendido2 . Click
do call BluetoothClient1 . SendText
text "4"
set BT_encendido2 . Enabled to false
set BT_apagado2 . Enabled to true
set BT_intermitente2 . Enabled to true
```

Enviar determinado dato al módulo Bluetooth de acuerdo con el botón presionado

(d) Componentes de bloques App Inventor



Figura 3. 11 Proceso de elaboración del control de luces



```

do
  to Frecuencia_Calorias
  do
    if (get global velocidad >= 0 and get global velocidad < 13)
    then
      set global carga_aerobica to 29.5
      set global aerobico_vo to 15
    else if (get global velocidad >= 13 and get global velocidad < 19)
    then
      set global carga_aerobica to 42
      set global aerobico_vo to 26
    else if (get global velocidad >= 19 and get global velocidad < 25.5)
    then
      set global carga_aerobica to 56.1
      set global aerobico_vo to 37
    else if (get global velocidad >= 25.5 and get global velocidad < 32)
    then
      set global carga_aerobica to 74
      set global aerobico_vo to 58
    else if (get global velocidad >= 32 and get global velocidad < 40)
    then
      set global carga_aerobica to 93.6
      set global aerobico_vo to 79
    else
      set global aerobico_vo to 100
      set global carga_aerobica to 100
    set global frecuencia to (get global frec_max * get global carga_aerobica) / 100
    set global aerobico_vo2 to (get global vo2 * get global aerobico_vo) / 100
    set global calorias to (get global calorias + (get global aerobico_vo2 * get global peso) / 12000)
  end do
end do
  
```

Cálculo de frec. cardiaca, carga aeróbica y calorías quemadas de acuerdo con la velocidad

(d) Componentes de bloques App Inventor

Figura 3.12 Proceso de obtención de parámetros de rendimiento físico



(a) Vista previa App Inventor

(b) Gráfica Pulso cardiaco

```

to graficarPunto y
do
  initialize local x to
  + get global xAnt + 2
  in call Grafica DrawLine
    x1 get global xAnt
    y1 get global yAnt
    x2 get x
    y2 get y
  set global yAnt to get y
  if get x >= Grafica Width
  then call Grafica Clear
    set global xAnt to 0
  else set global xAnt to get x
  
```

Graficar pulso de acuerdo con los valores recibidos via Bluetooth

```

when Screen10 Initialize
  Screen starting mpo_Grafica TimerEnabled to false
  set global factorEscala to Grafica Height / 180
  set BluetoothClient1 DelimiterByte to 10
  set Grafica BackgroundColor to black
  set Grafica PaintColor to green
  
```

Configurar propiedades de la gráfica

(c) Componentes de bloques App Inventor



(d) Componentes de diseño App Inventor

Figura 3. 13 Proceso de obtención de gráfica de pulso cardiaco

3.2 CONEXIÓN BLUETOOTH

En primer lugar, se realiza la conexión del módulo Bluetooth HC-05 a los pines de la plataforma Arduino con la finalidad de realizar su configuración como elemento maestro-esclavo del sistema, es decir, para que sea el elemento que recibe y a su vez emite las instrucciones de parte de la app y de la plataforma Arduino respectivamente.

Tal y como muestra la *figura 3.14*, los pines de transmisión y recepción (Tx y Rx) respectivamente, son conectados a los pines del Arduino asignados para llevar a cabo la comunicación serial con el módulo HC-05.

Así mismo, el pin KEY o ENABLE del HC-05 es inducido a un pin en estado alto de la plataforma Arduino durante la inicialización del sistema, de esta forma se garantiza que el módulo entre en un estado de configuración, con lo que ahora es posible la modificación de sus parámetros a través de comandos desde el monitor serial integrado en la IDE de Arduino.

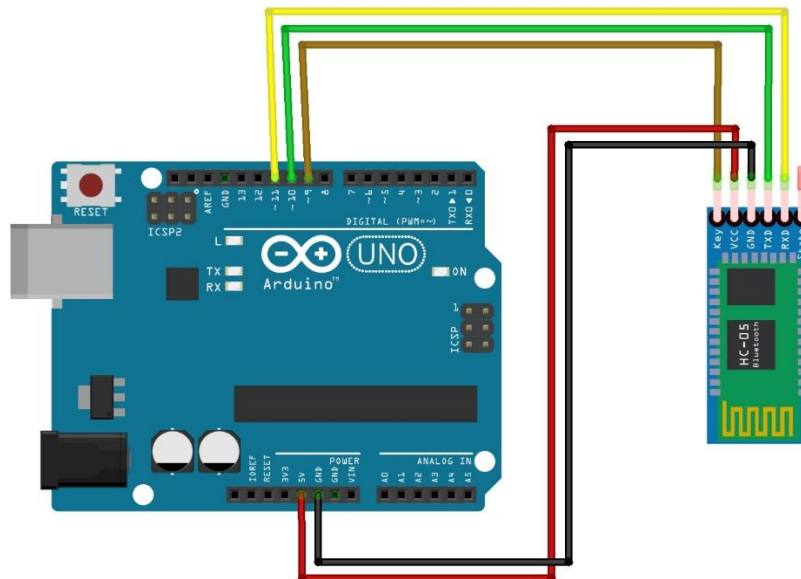


Figura 3. 14 Diagrama de conexión del módulo HC-05 y la plataforma Arduino

Habiendo establecido la conexión física, entre el Arduino y el módulo HC-05 se procede a cargar al microcontrolador de la plataforma Arduino Uno, el sketch que hace posible la comunicación de forma serial entre la plataforma Arduino y un ordenador personal, y a su vez con el módulo Bluetooth HC-05. Dicho proceso se realiza a través del software de Arduino, instalado en una computadora. Las sentencias de código que conforman el sketch se muestran en la *figura 3.15*



```

#include<SoftwareSerial.h>      // Librería embebida (Comunicacion Serial)
SoftwareSerial BTSerial(10,11); // Pines 10 y 11 -rx, TX Bluetooth

void setup() {
  pinMode(9, OUTPUT);          // Pin 9 Salida Digital
  digitalWrite(9, HIGH);
  Serial.begin(9600);          // velocidad comunicacion Ordenador - Arduino (bits/s)
  Serial.println("Ingrese el comando AT:"); // Induce al usuario a introducir comandos
  BTSerial.begin(38400);       // velocidad comunicacion Bluetooth - Arduino (bits/s)
}

void loop() {
  // Muestra en monitor serial de Arduino los datos contenidos en módulo BT

  if (BTSerial.available())
    Serial.write(BTSerial.read());

  // Muestra en módulo BT los datos contenidos en el monitor serial de Arduino

  if (Serial.available())
    BTSerial.write(Serial.read());
}

```

Figura 3. 15 Sketch destinado a establecer una comunicación serial con la PC y el HC-05

Teniendo una comunicación serial establecida, se configura el módulo HC-05 a través de los siguientes comandos a través del monitor serial de Arduino.

Tabla 3. 1 Comandos para la configuración del módulo bluetooth HC-05

COMANDO	RESPUESTA HC-05
AT	Retorna "OK"
AT + ORGL	Restaura valores de fábrica
AT + NAME = ...	Cambiar nombre
AT + PSWD	Cambiar contraseña
AT + ROLE	Establece el rol (1=Maestro, 0=Esclavo)
AT + ADDR	Muestra dirección MAC
AT + RESET	Resetea el módulo

3.3 SKETCH ARDUINO

La configuración inicial del módulo HC-05 tiene la finalidad de habilitar la comunicación serial entre la plataforma Arduino y el módulo HC-05, y a su vez la comunicación de este último con la App. Es decir, El módulo Bluetooth HC-05 básicamente realiza la función de intermediario, comunicando la aplicación instalada en el smartphone con la plataforma Arduino.

En este proyecto específicamente, son dos las funciones que implican la transmisión y recepción de datos desde la plataforma Arduino UNO: El control del sistema de luces y la gráfica del pulso cardiaco del usuario.

Para ello es indispensable crear los eventos correspondientes en del código principal dentro de la App para generar, a través de los diferentes botones y demás componentes de diseño, las señales necesarias para interactuar con el código cargado en el microcontrolador de la plataforma Arduino, dicho código es conocido como “sktech”.

El sketch cargado a la plataforma Arduino que hace posible que por un lado el usuario determine el estado de las salidas del circuito por medio de botones y por otro permite leer valores de muestreo de una señal analógica de entrada y a su vez graficar dicha señal a través de la app se muestra en la *figura 3.16*.



```
Pulso_cardiaco_luces_bt$

int Pulso=0;
int Salida=11;
int Red=10;
boolean statusLedRed = true;
int ld=8;
int lt=9;
int estado=0;

void setup() {

  Serial.begin(9600);           // Velocidad comunicación con Ordenador (bits/s)
  pinMode(A0, INPUT);         // A0 - Entrada Analógica (Sensor pulso)
  pinMode(Salida, OUTPUT);    // Pin 11 - Salida digital (LED pulso)
  pinMode(ld, OUTPUT);        // Pin 8 - Salida digital (Luz delantera)
  pinMode(lt, OUTPUT);        // Pin 9 - Salida digital (Luz trasera)
  //Serial1.begin(57600);      // Velocidad comunicacion con modulo Bluetooth HC-05

}
```



```
✓ → 🏠 ⬆️ ⬇️
Pulso_cardiaco_luces_bt$
void loop() {

  Serial.print("#");      // Símbolo de Inicio de línea "#"
  Pulso=analogRead(A0);   // Variable Pulso = Valor analógico de A0

  //***** LECTURA DE SENSOR DE PULSO *****

  if (Pulso>=520)        // Determina cuando hay un pico en el pulso
  {
    digitalWrite(Salida,HIGH);
    delay(30);
    digitalWrite(Salida,LOW);
  }
  delay(50);
  Serial.println(Pulso);  // Envía Inf, Pulso cada 50ms

  //***** LECTURA BOTONES LUCES *****

  if(Serial.available(>0){
    estado = Serial.read();
  }
  if (estado == '1'){
    digitalWrite(ld,HIGH);
  }
  if(estado=='2'){
    digitalWrite(ld,LOW);
  }
  if (estado == '3'){
    digitalWrite(lt,HIGH);
  }
  if(estado=='4'){
    digitalWrite(lt,LOW);
  }
  if (estado == '5'){
    digitalWrite(ld,HIGH);
    delay(500);
    digitalWrite(ld,LOW);
    delay(500);
  }
  if(estado=='6'){
    digitalWrite(lt,HIGH);
    delay(500);
    digitalWrite(lt,LOW);
    delay(500);
  }
}
```

Figura 3. 16 Sketch para hacer la comunicación entre la plataforma Arduino y módulo HC-05

3.4 SISTEMA DE ILUMINACIÓN

Una vez ya configurados los dispositivos que constituyen la parte lógica y de comunicación del sistema (Plataforma Arduino, App y módulo HC-05) es necesario vincularlos de algún modo de tal forma que puedan interactuar entre ellos.

Para esto, se procede a crear un evento dentro del código de la app que asigne a un botón dentro de la interfaz gráfica la función de mostrar la lista de dispositivos con bluetooth disponibles a través de su dirección MAC, como se muestra en la *figura 3.17*, y una vez seleccionado, establecer la conexión Bluetooth entre el Smartphone y el módulo HC-05, los bloques necesarios para realizar dicha operación en la plataforma de MIT App Inventor se muestra en la *figura 3.9*.

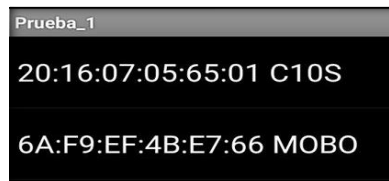


Figura 3.17 Lista de dispositivos bluetooth vinculados con el Smartphone

Una vez habilitada la conexión bluetooth a través del emparejamiento de dispositivos, para prender y apagar las luces delantera y/o trasera, el bloque de función correspondiente en la app se puede apreciar en la *figura 3.9 (d)*. A través de un botón de la App se genera un digito de identificación el cual será recibido por la plataforma Arduino, esta a su vez identificara la solicitud y gestionará los recursos del sistema que sean necesarios, en este caso cambiara el estado de un pin de salida del microcontrolador a 0 o 1 dependiendo de la solicitud del usuario, esta etapa es controlada a través del sketch mostrado en la figura 3.16.

El funcionamiento básico del sistema descrito anteriormente se puede apreciar en las siguientes *figura 3.18* en donde se prueba el encendido y apagado de luces haciendo uso de un circuito de prueba montado sobre un protoboard..

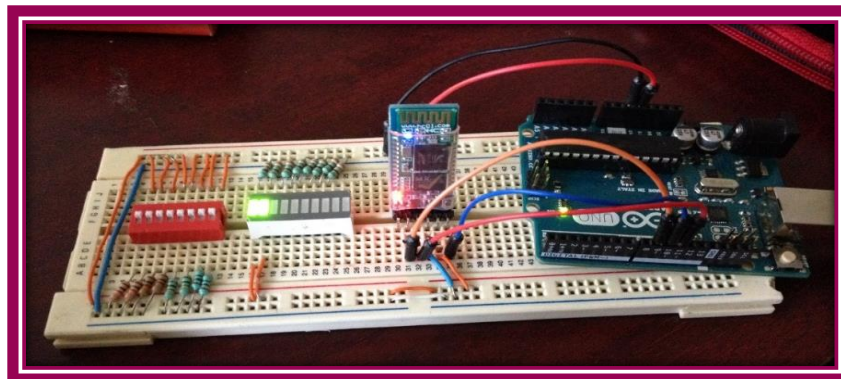


Figura 3.18 Prueba de encendido de luz delantera y luz trasera

3.5 GRÁFICA DE PULSO CARDIACO

Al igual que en el sistema de iluminación, lo primero que debe considerarse al momento de graficar la señal del pulso cardiaco es que deben estar vinculados a través de una conexión Bluetooth el Smartphone y el módulo HC-05,

Sin embargo, el orden de la comunicación entre los dispositivos del sistema es inverso al caso del control de luces ya que ahora es la plataforma Arduino quien a través del sensor APS 9008 recibe y procesa la información directamente del usuario para después enviarla a la App a través del módulo Bluetooth.

El sketch que permite la obtención de la información de parte del sensor del pulso cardiaco viene dado en la figura 3.16. Dentro de este código se aprecia un muestreo de información cada 50 ms con lo cual es posible la construcción de una gráfica apegada al comportamiento del ritmo cardiaco del usuario.

Dicha información, después de ser recibida y reenviada por la plataforma Arduino, lo mismo que por el módulo HC-05, llega al Smartphone, concretamente a la App que a través de los bloques de función mostrados en la figura ,, es capaz de extrapolar los datos recibidos en una gráfica que el usuario del sistema puede interpretar sin mayor problema.

En la figura 3.19 se muestra la interacción del usuario con el sensor APS 9008 y la plataforma Arduino sobre un circuito de prueba en protoboard. Por otra parte en la figura 3.20 se muestra el resultado del muestreo de l información de pulso cardiaco del usuario a través del monitor serial de la plataforma Arduino, en donde se pueden observar los picos de señal por encima de los 520

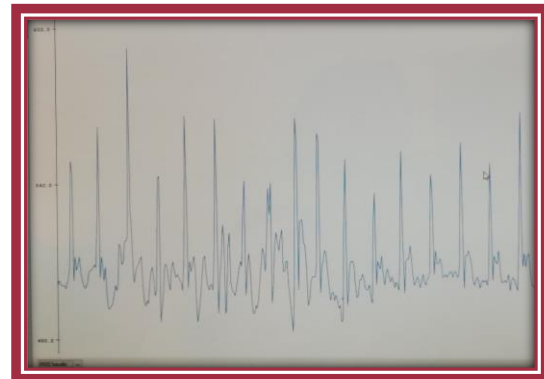
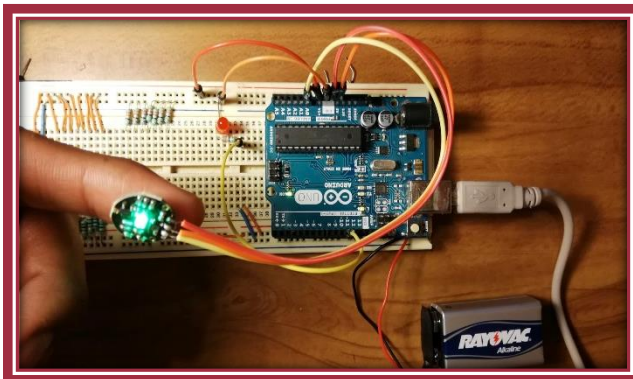
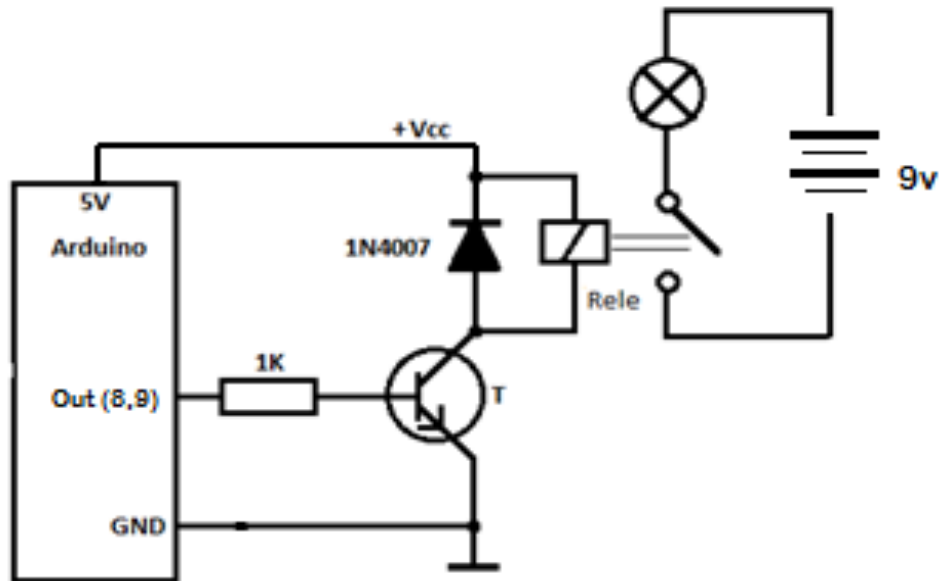


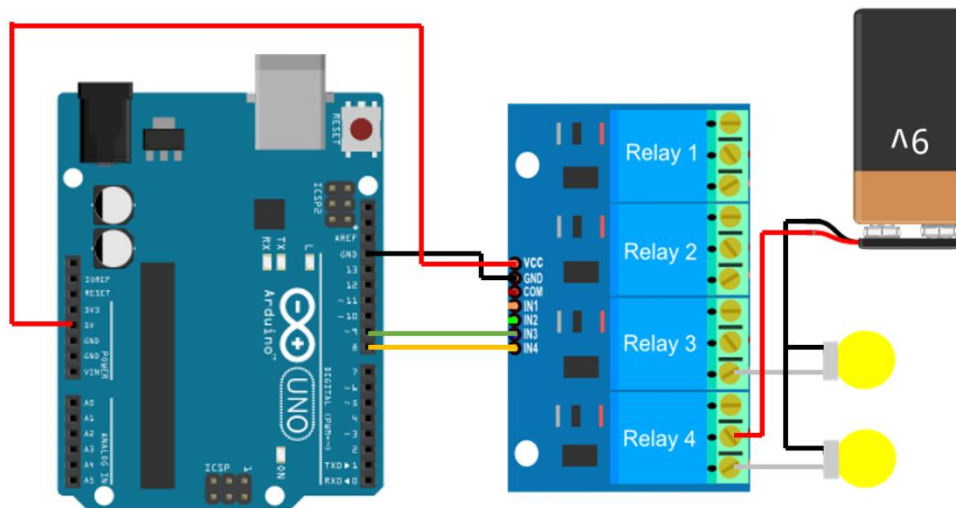
Figura 3. 19 Circuito de prueba de pulso cardiaco Figura 3. 20 Gráfica de pulso cardiaco Arduino

3.6 MODULO DE RELEVADORES

Una vez habiendo perfeccionado la lógica del control de luces dentro del proyecto, es necesario implementar una etapa de potencia que proporcione a nuestro circuito la capacidad de alimentar, simultáneamente, los dispositivos finales del circuito. Para ello se emplea un circuito basado en elementos conmutadores, en este caso, relevadores. El diseño del circuito para el control de luces se muestra en la siguiente figura:



a) Diagrama eléctrico, control de una luz



b) Diagrama pictórico, control de luces

Figura 3.21 Control de luces a través de la plataforma Arduino

3.7 MODULO DE CONTROL (CIRCUITO)

Relacionando las distintas etapas de hardware del sistema de iluminación y del sistema para graficar el pulso cardiaco da como resultado el circuito que se muestra a continuación en la figura 3.22.

En primer lugar, es posible inferir la función de la plataforma Arduino como elemento central del módulo, es la encargada del control y la gestión de los recursos, así como transmisión de datos en base a la información recibida por parte de la App a través del módulo Bluetooth HC-05 y por parte del sensor del pulso cardiaco.

La etapa final está constituida por los componentes de potencia (Relevadores), los cuales hacen posible la energización de los dispositivos terminales de salida (Luces) a través de su función de switcheo, con lo cual es posible alimentarlas a través de una batería adicional de 9 volts, generadora de una corriente suficiente (1000mA/h), tomando en consideración el consumo de las luces.

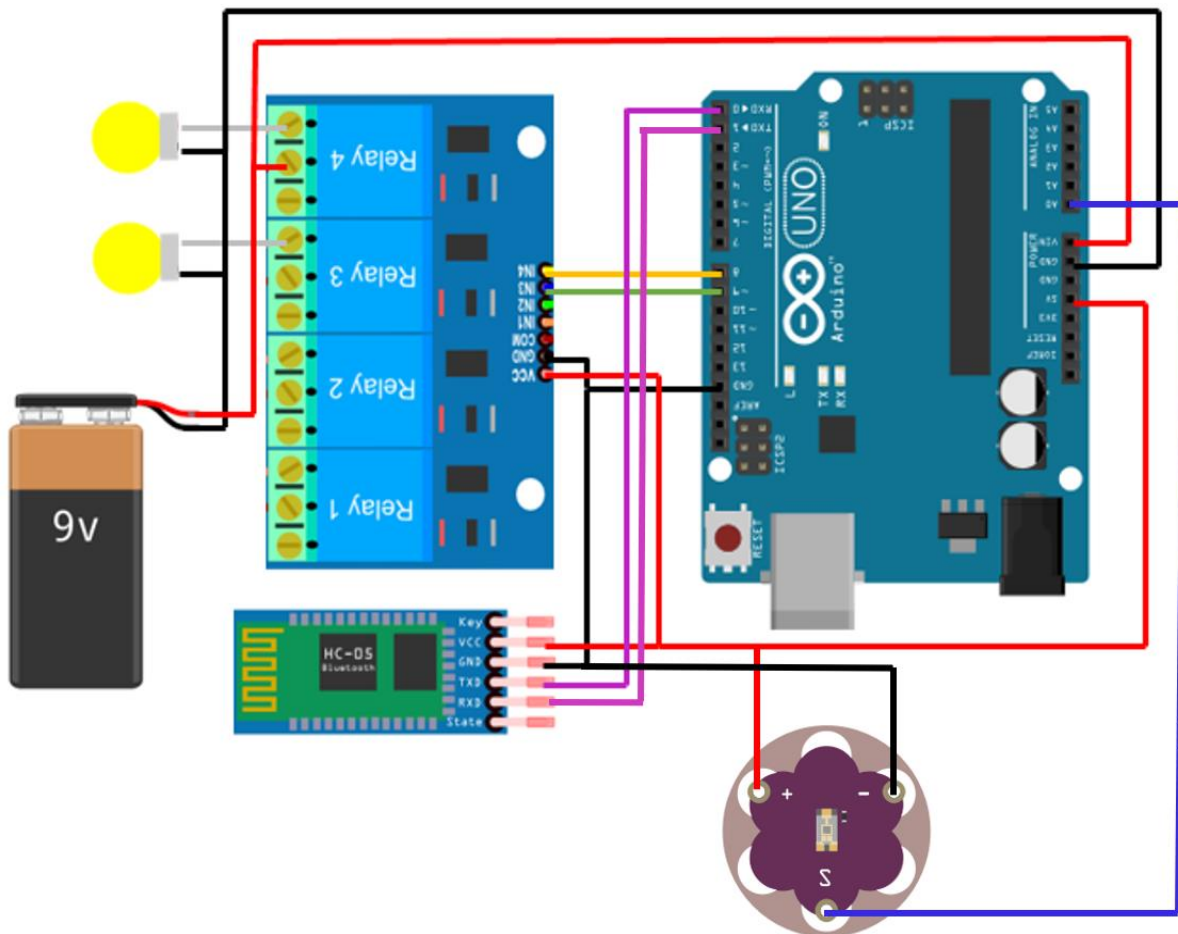
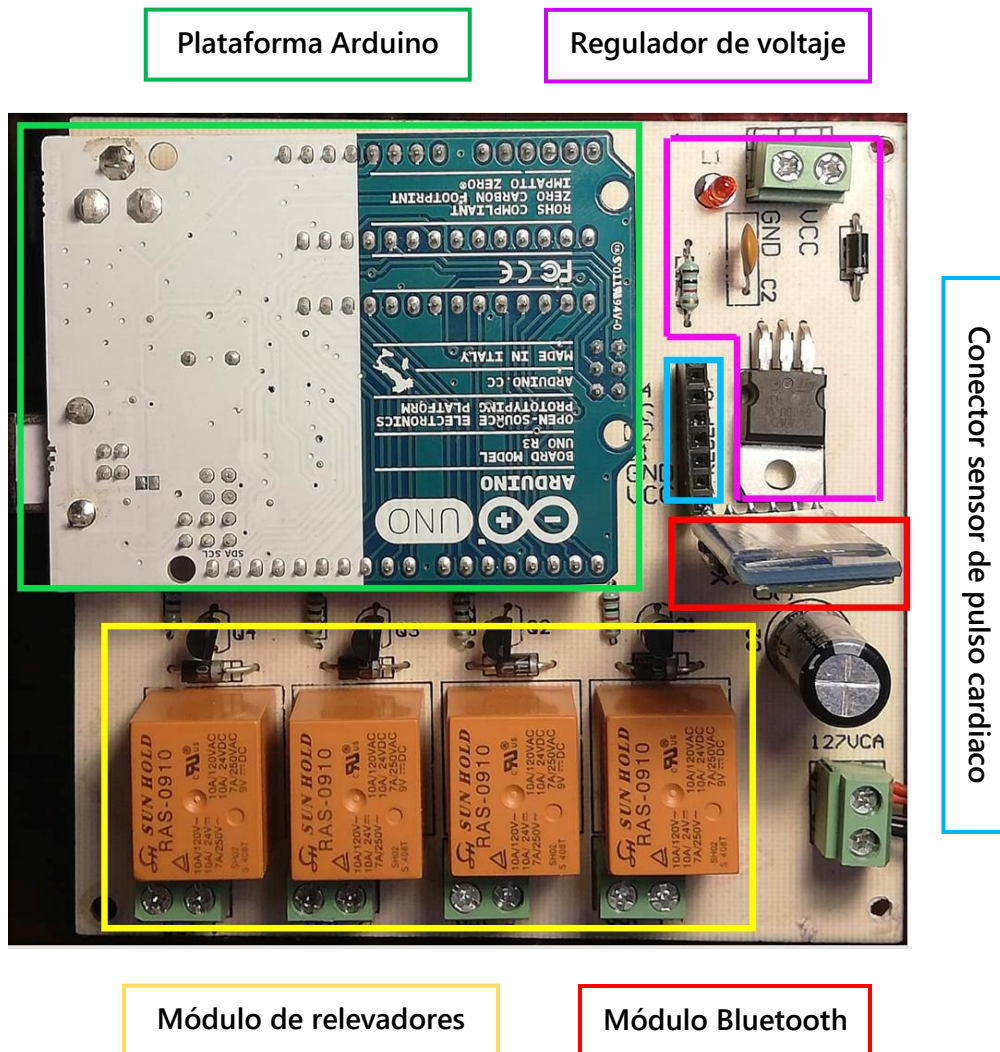


Figura 3. 22Diagrama de módulo de control de sistema para bicicleta

Tomando en cuenta todos los dispositivos y componentes que intervienen en el proyecto y la interacción que tienen entre ellos se diseña un circuito que concentra todos los elementos de la forma mas compacta posible, tal como se muestra en la figura 3.23.



Plataforma Arduino

Regulador de voltaje

Conector sensor de pulso cardiaco

Módulo de relevadores

Módulo Bluetooth

Figura 3.23 Circuito de módulo de control de sistema para bicicleta

El circuito se construye sobre una tabla fenólica de cobre de 10X10 cm, sobre la cual se planchan las pistas de cobre que unen los diferentes componentes, como se muestra en la Figura...

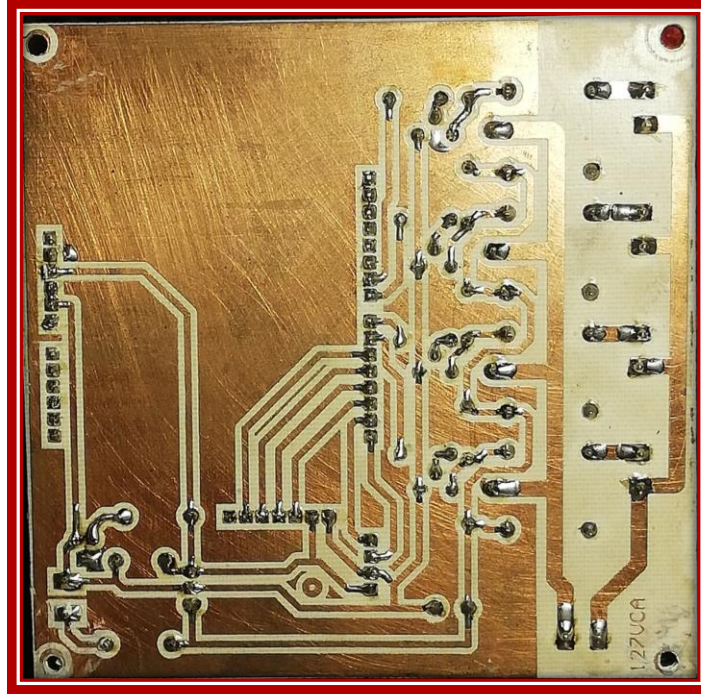


Figura 3.24 Placa de circuito impreso

Una vez que se ha perforado y soldado los componentes sobre la placa quedando como se muestra en la Figura....



Figura 3.25 Vista lateral circuito de módulo de control



3.8 CÁLCULO DE PARÁMETROS DE RENDIMIENTO FÍSICO

En primer lugar, es necesario establecer que los parámetros de rendimiento están en relación directa con la velocidad ejercida por el usuario a lo largo de su recorrido, por ello, el primer dato que debe obtenerse es la velocidad, cuyos componentes son el tiempo y la distancia.

El tiempo es medible a través de un cronometro, función disponible dentro de cualquier Smartphone a través de sus sensores internos. Mientras tanto, para medir la distancia recorrida a partir de dos puntos de geolocalización es necesario emplear la fórmula de Haversine, cuyo modelo matemático se plasma en la *ecuación 3.1* [25].

$$Distancia = 2 * R * asin \sqrt{\sin^2 \left(\frac{\Delta lat}{2} \right) + \cos(lat1) * \cos(lat2) * \sin^2 \left(\frac{\Delta lon}{2} \right)}$$

Ecuación 3. 1 Fórmula de Haversine

En donde la R hace referencia al radio de la tierra (61000 m), mientras que las longitudes y latitudes están en función de la ubicación del usuario en dos instantes de tiempo diferentes.

Una vez obtenida la velocidad, dividiendo la distancia entre el tiempo, y haciendo uso del modelo matemático de Tanaka (ecuación 3.2), es posible obtener la carga aeróbica empleada en función de su frecuencia cardiaca máxima [26].

$$Frecuencia \left\{ \begin{array}{l} \text{Hombres: } 208.7 - (0.73 * Edad) \\ \text{Cardiaca máx} \left\{ \begin{array}{l} \text{Mujeres: } 208.1 - (0.77 * Edad) \end{array} \right. \end{array} \right.$$

Ecuación 3. 2 Fórmula de Tanaka para obtener la FC máx

Habiendo hecho la estimación del porcentaje de la frecuencia cardiaca empleada por el usuario, es necesario relacionarla con el porcentaje de consumo máximo de oxígeno VO₂ máx. Existe una relación lineal entre estos dos parámetros, como se muestra en la siguiente tabla 3.2.

Tabla 3. 2 Relación de porcentajes entre la FC máx empleada y el VO2 máx consumido

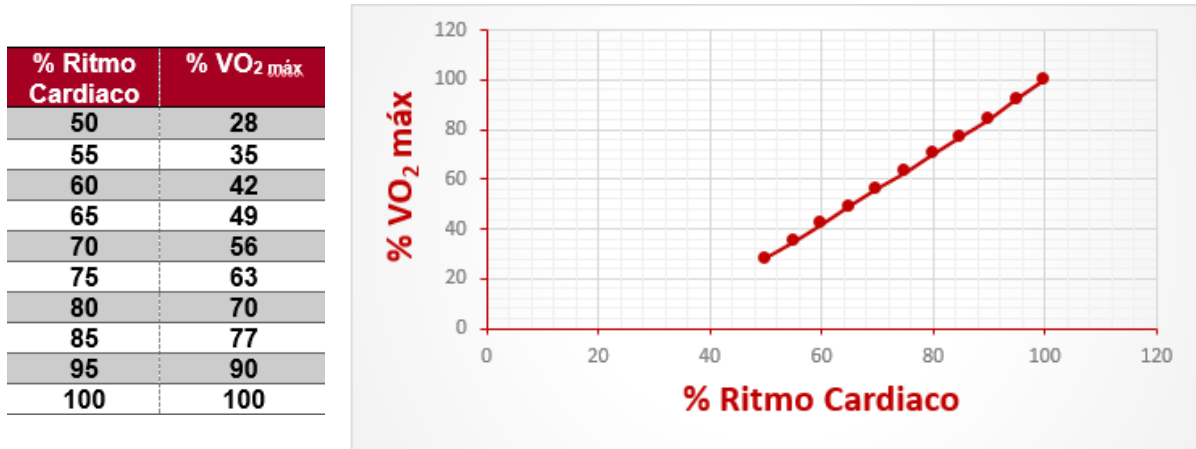


Tabla 3. 3 niveles de VO2 máx (ml/kg.min) en hombres

Edad	Pobre	Promedio	Bueno
13-19	<35	35-50.9	>51.0
20-29	<33	33-46.4	>46.5
30-39	<31.5	31.5-44.9	>45.0
40-49	<30.2	30.2-43.7	>43.8
50-59	<26.1	26.1-40.9	>41.0
60+	<20.5	20.5-36.4	>44.3

Tabla 3. 4 Niveles de VO2 máx (ml/kg.min) en mujeres

Edad	Pobre	Promedio	Bueno
13-19	<25.0	25.0-38.9	>39.0
20-29	<23.6	23.6-36.9	>37.5
30-39	<22.8	22.8-35.6	>35.7
40-49	<21.0	21.0-32.8	>32.9
50-59	<20.2	20.2-31.4	>31.5
60+	<17.5	17.5-30.2	>30.3

Finalmente, obtenemos las calorías quemadas a partir del VO2 máx haciendo uso del siguiente modelo matemático recomendado por el Colegio Americano de Medicina del Deporte (ACSM) [27].

$$\text{Calorías/min} = \frac{\text{Peso corporal} * \text{VO}_2}{200}$$

Ecuación 3. 3 Formula del ACSM para la estimación de calorías



3.9 DIAGRAMA DE FLUJO

El proceso de funcionamiento del proyecto comienza con el encendido del módulo de control, alimentado por una batería LiPo de 9V que alimenta la etapa de control, específicamente el Arduino UNO, que a su vez alimenta el módulo Bluetooth, y el módulo de relevadores.

Posteriormente se habilitan las conexiones Bluetooth en el Smartphone y se lleva a cabo el inicio de la App por parte del usuario desplegando así la pantalla de arranque (Splash screen),

Después, se muestra la pantalla de Iniciar Sesión donde se podrá ingresar si es que se cuenta con un usuario y contraseña, de no ser así se muestra la opción de Registro, una vez realizado el registro se ingresará y se podrá iniciar el recorrido. Es importante mencionar que en este punto el usuario ya debió haber colocado su dedo sobre el sensor de pulso cardíaco que viene implementado en la bicicleta.

Una vez iniciado el recorrido se presenta un menú con tres opciones a elegir: Seguimiento de ruta, Control de luces y Parámetros de rendimiento físico.

En la opción de seguimiento de ruta se muestra en pantalla el mapa de geolocalización en tiempo real, pudiendo expandirlo a la pantalla completa. Así mismo, proporciona información relacionada al recorrido, como lo son distancia, tiempo y velocidad promedio del recorrido.

En la opción de Control de luces se realiza la vinculación Bluetooth entre el módulo HC-05 y el dispositivo móvil, en pantalla se muestran los botones correspondientes para el control de las luces delantera y/o trasera, poniéndolas en encendido, apagado o intermitente según las necesidades del usuario.

En la opción de Parámetros de rendimiento físico se muestra una estadística sobre el rendimiento que ha tenido el usuario, con los datos obtenidos durante el recorrido, tales como: la carga aeróbica, las calorías quemadas y la frecuencia cardíaca. Además, se presenta la opción al usuario a través de un botón, de monitorear el comportamiento de su pulso cardíaco a través de una gráfica a pantalla completa.

Finalmente se presenta la opción de finalizar sesión con lo cual se da por concluido el recorrido, procediendo con la desconexión de los dispositivos Bluetooth y el apagado del módulo de control.

El funcionamiento del sistema se ve plasmado en el diagrama de flujo presentado a continuación en la figura 3.26.

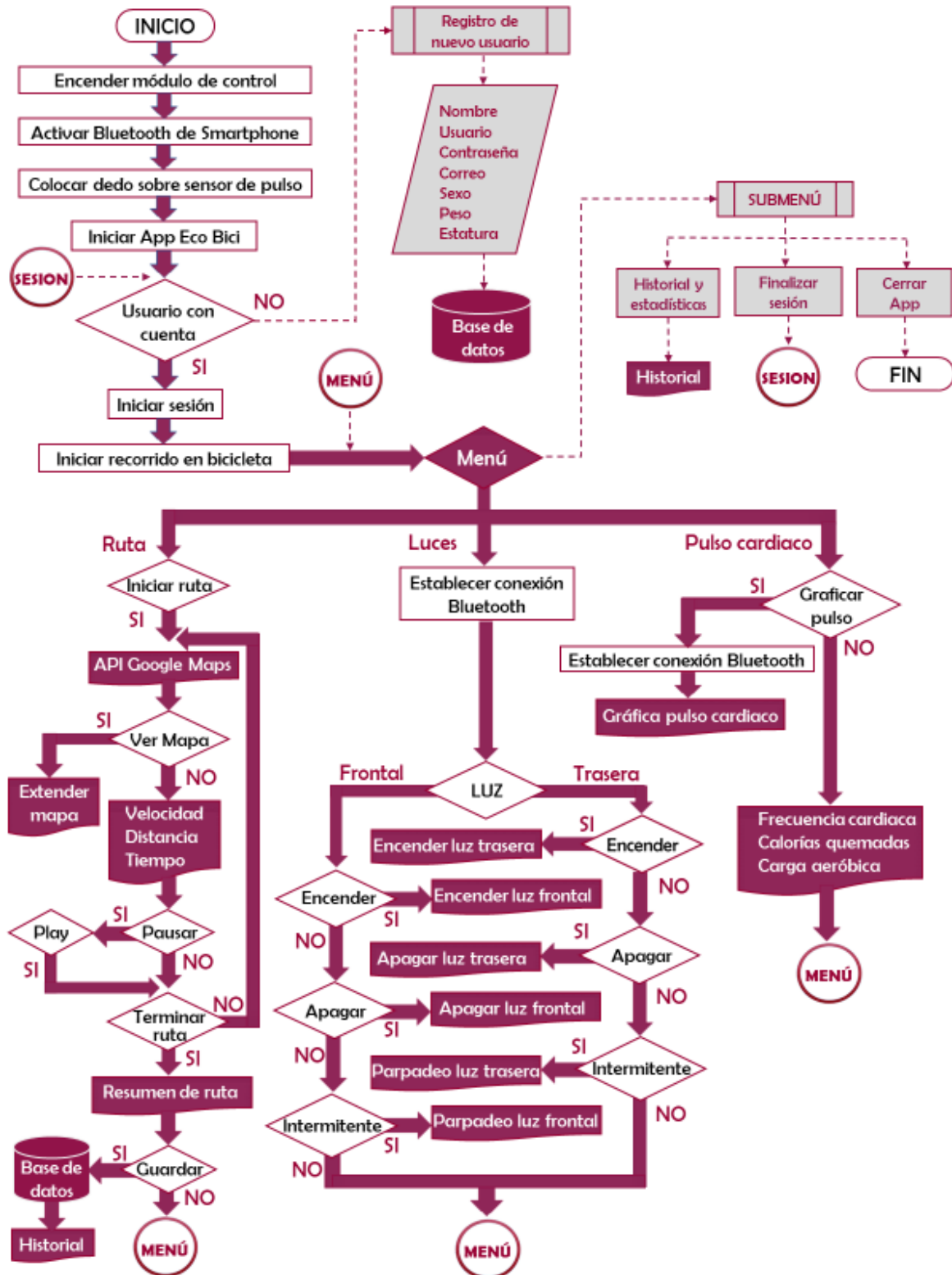


Figura 3. 26 Diagrama de flujo del sistema Eco Bici



CAPÍTULO 4



PRUEBAS Y RESULTADOS



4

PRUEBAS Y RESULTADOS



El siguiente capítulo consiste en explicar la parte final del trabajo, la evaluación y corrección del proyecto, el capítulo básicamente trata los siguientes puntos:

- ❖ El proceso de compilación, descarga e instalación de la App desde la plataforma MIT App Inventor 2 hasta el smartphone.
- ❖ El manejo de la App una vez alojada en el Smartphone, sus diferentes interfaces con sus respectivas funciones, así como la forma de emplearlas.
- ❖ Los diversos problemas e incongruencias detectados en las primeras versiones de la App, así como su depuración y solución dentro del sistema.
- ❖ Una descripción detallada del proceso de implementación del sistema sobre la bicicleta, sus componentes, las fallas corregidas y las pruebas de funcionamiento correspondientes.

4.1 APP

4.1.1 Descarga de la aplicación móvil

Una vez terminado el desarrollo de las distintas secciones que conforman el proyecto, se realizan las pruebas pertinentes en función de evaluar el desempeño de la app y su interacción con las demás partes, así como la modificación y optimización los componentes que lo requieran.

Para llevar a cabo las pruebas del proyecto, primeramente, se lleva a cabo la compilación del programa directamente desde la plataforma de MIT App Inventor 2, el resultado es un archivo en formato .apk que contiene el programa ahora en código ejecutable.

También existe la posibilidad de ejecutar un simulador de Smartphone dentro de la plataforma de App Inventor, para facilitar la observación de los avances y modificaciones al código.

En la *figura 4.1* se ilustra el proceso de compilación y descarga de la app desde la plataforma App Inventor:

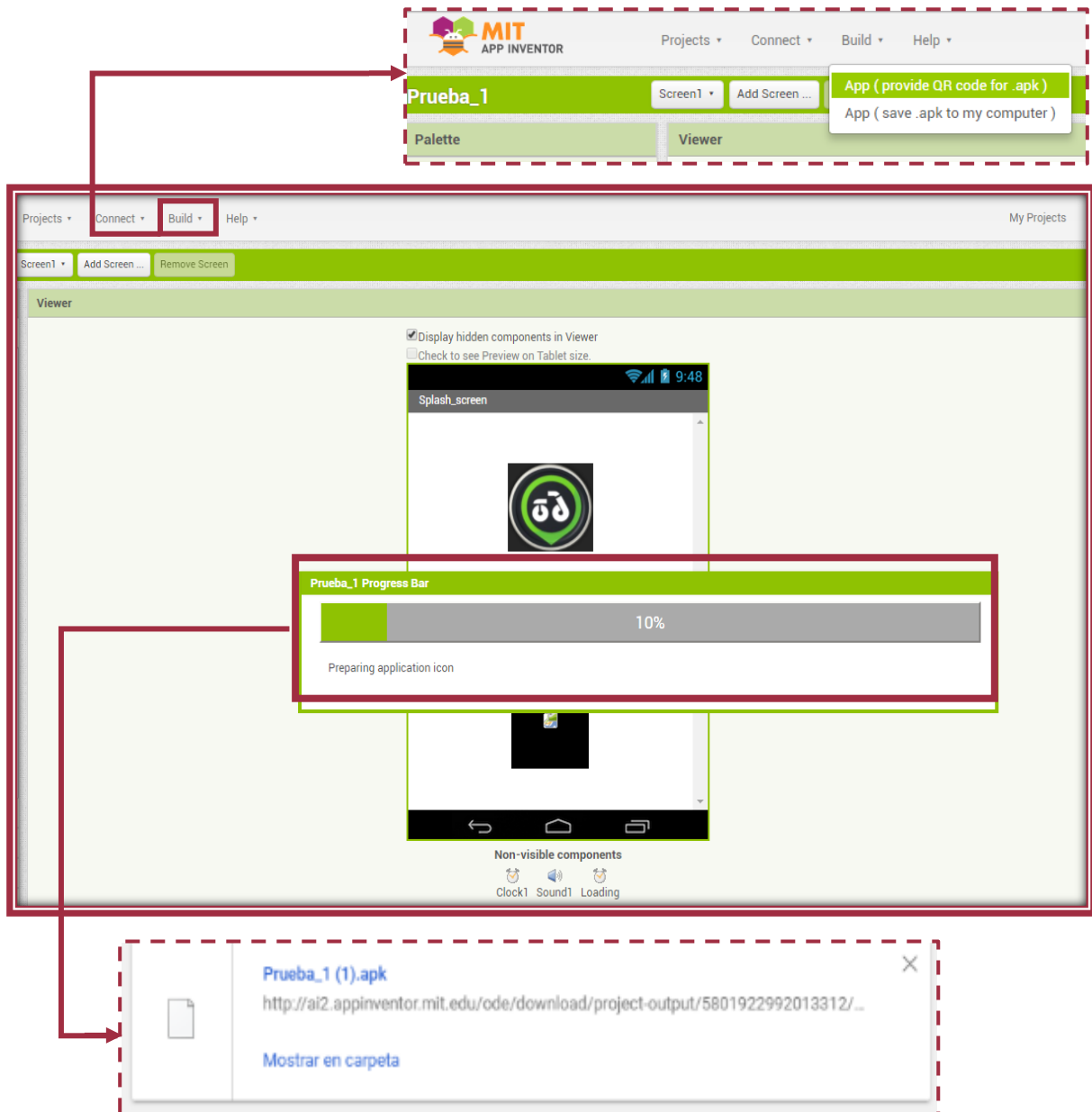


Figura 4.1 Compilación y descarga de App desde la plataforma App Inventor 2

Habiendo descargado el archivo .apk, se transfiere desde la computadora hacia el Smartphone, en donde se alojará la App, ya sea de forma alámbrica, a través de correo o por tecnología Bluetooth. Una vez alojado, el archivo dentro del smartphone se ejecuta el proceso de instalación de la App.

4.1.2 Ícono y pantalla de inicio

Después de los diferentes cambios de diseño de interfaz, se llega al diseño final de la app, cuyas principales funciones se muestran a continuación a través de capturas de pantalla tomadas directamente de la app previamente instalada dentro del menú de aplicaciones del smartphone, como se muestra en la *figura 4.2*.

Cabe mencionar que el programa consiste en una aplicación nativa dentro del sistema operativo Android, es decir, para su funcionamiento emplea los recursos de software y hardware disponibles dentro del smartphone.

Una vez desplegada la pantalla de inicio del Smartphone, se selecciona la aplicación con el logotipo característico de la app a utilizar para empezar con el uso de la misma.

La primera interfaz consiste en mostrar el logo y nombre la aplicación, además de contener un splash screen que denota el proceso de carga de la app durante un lapso corto de tiempo, acompañado de un efecto de sonido.

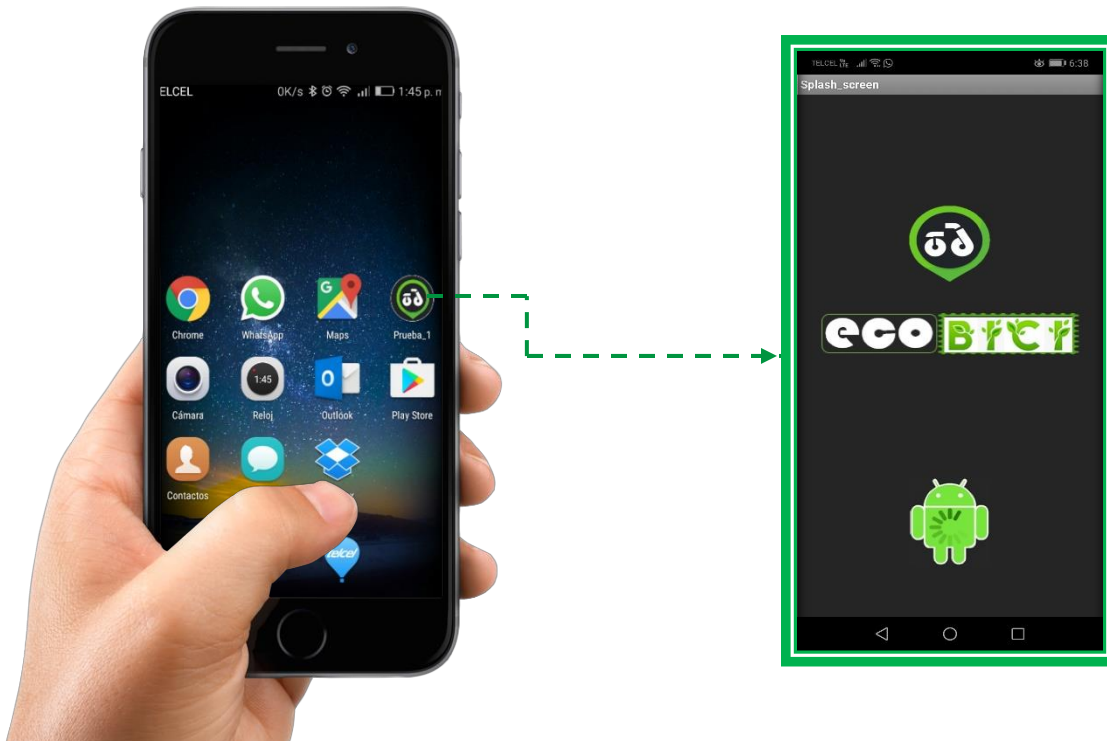


Figura 4. 2 Ícono de la app y Splash Screen

4.1.3 Registro y validación de usuario

En la *figura 4.3* es posible apreciar el proceso de validación de un usuario a través de una contraseña. En caso de no contar con una cuenta personal, se tiene la opción de crear una, llenando un registro con datos básicos. Dicha información permanece almacenada confidencialmente en una base de datos en la memoria no volátil del Smartphone, para ser utilizada posteriormente por la app.

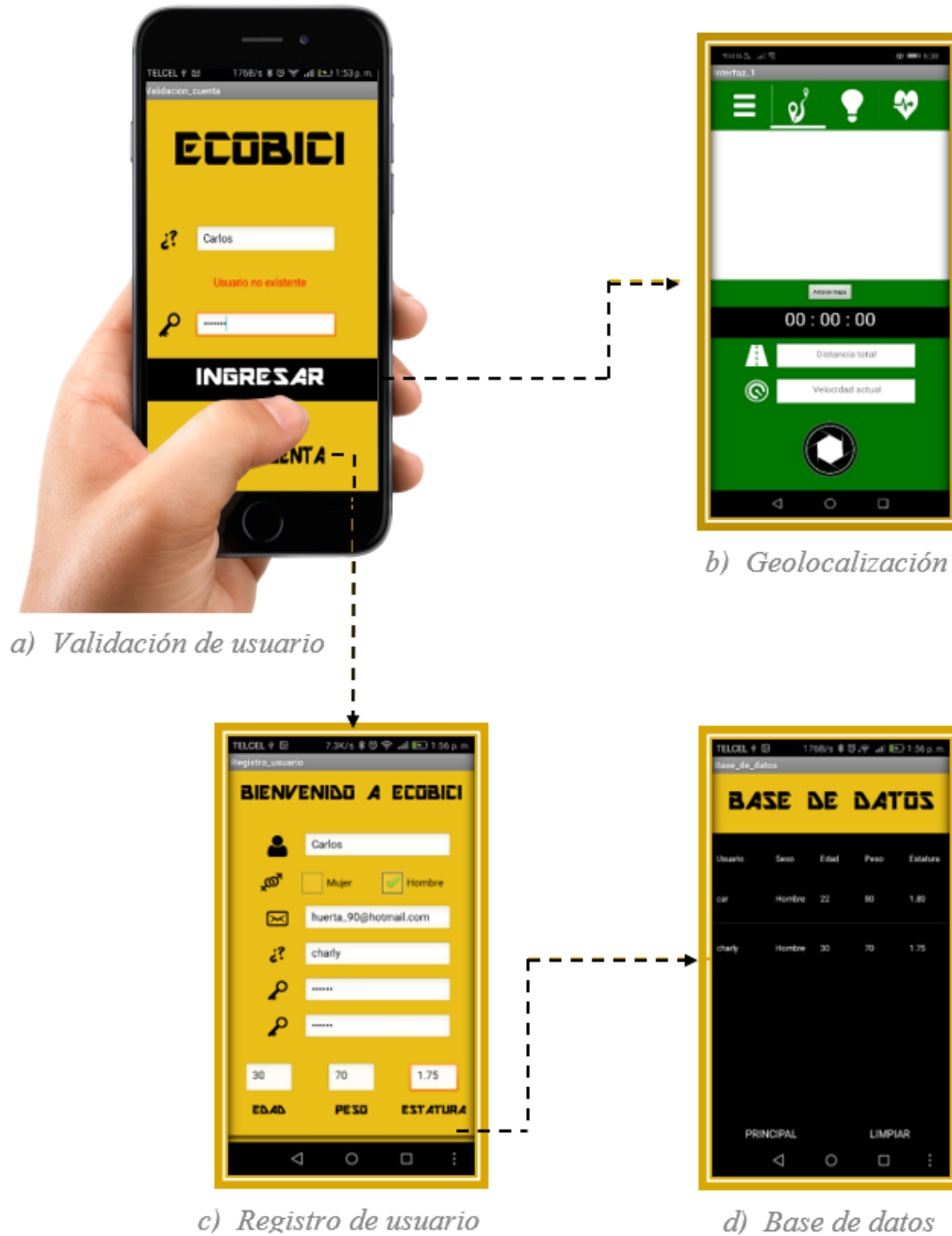


Figura 4.3 Proceso de registro y validación de cuenta del usuario

La primera interfaz, consiste básicamente en dos cajas de texto (Text Box), donde el usuario ingresa sus datos (usuario y contraseña), los cuales son vinculados con la base de datos de la app y, una vez que son ingresados al sistema a través de un botón, son validados de acuerdo con los usuarios previamente registrados por medio de una comparación de cadenas de caracteres, como se muestra en la *figura 4.4*

En caso de ingresar alguna información que no corresponda con los elementos almacenados en la base de datos, se muestra una notificación. La app no pasará a la siguiente interfaz o screen, hasta que los datos ingresados sean correctos o se cree una nueva cuenta de usuario por medio del botón correspondiente.

En esta interfaz, al tratarse básicamente de una comparación entre dos registros de la base de datos del Smartphone y dos cadenas de texto ingresadas por el usuario, no se presentaron obstáculos mayores en el desarrollo. Cabe mencionar que actualmente la validación no diferencia entre letras mayúsculas y minúsculas.

Una vez que se elige la opción de crear una cuenta la aplicación cambiara de pantalla a una en la cual el usuario pueda llevar a cabo ese registro.



Figura 4.4 Componentes de la interfaz "Validación de usuario"

Una vez que se elige la opción de crear una cuenta, se llega a la interfaz mostrada en la *figura 4.5* Conformada igualmente por un conjunto de cajas de texto, cuyo contenido es apilado y almacenado en la base de datos del sistema. Así mismo también se cuenta con un par de casillas de verificación (Check Box) para comprobar el sexo del usuario.

Cabe mencionar que el proceso de autenticación de correo, así como verificación de identidad de las personas a través de Captcha no se ha llevado a cabo debido a la complejidad de su implementación, siendo uno de los principales aspectos a mejorar en aras de optimizar el sistema.

La información es importante para ubicar al usuario en un perfil y de acuerdo a sus particularidades adaptar los algoritmos para que sean lo más apegados a la realidad. Igualmente otro de los objetivos a implementar en la app es un sistema de cifrado de información con la finalidad de garantizar al usuario que los datos proporcionados no serán vulnerados



Figura 4. 5 Componentes de la interfaz "Registro de usuario"



La base de datos por su parte se encarga de almacenar y ordenar, en la memoria no volátil del sistema, la información ingresada por el usuario al sistema, dicha información es confidencial. Sin embargo, para efectos de muestra del funcionamiento de la app, se despliega una lista con los principales datos de los usuarios registrados en la app, como se muestra en la *figura 4.6*.

La interfaz contiene además dos opciones, la primera de ellas (Limpiar) se encarga de vaciar completamente el contenido de la base de datos, por lo que toda la información previamente registrada se pierde. La otra opción (Principal) nos dirige hacia la interfaz principal de la app, desde la cual se inicia la sesión.

Es importante recalcar que esta interfaz no está diseñada para ser parte de la app proporcionada al usuario, y solo ha sido implementada en esta fase beta de la aplicación para efectos prácticos y con la finalidad de validar que la información de los usuarios sea almacenada correctamente.

Otra área de oportunidad que se ha identificado en este punto, para ser implementado posteriormente, es la migración de la información de los usuarios de la memoria del teléfono hacia la nube, utilizando las medidas de seguridad necesarias. De esta forma el usuario podría continuar viendo su historial en varios dispositivos móviles.



Figura 4. 6 Componentes de la interfaz "Base de datos"

4.1.4 Geolocalización, distancia, tiempo y velocidad

La interfaz principal consiste en una serie de parámetros del recorrido (cronómetro, distancia total y velocidad actual), que se muestran al usuario en tiempo real una vez que se pulsa el botón de inicio. Así mismo, se muestra el punto de ubicación del usuario dentro un mapa de la zona en la cual se realiza el recorrido, como es posible apreciar en la *figura 4.7*

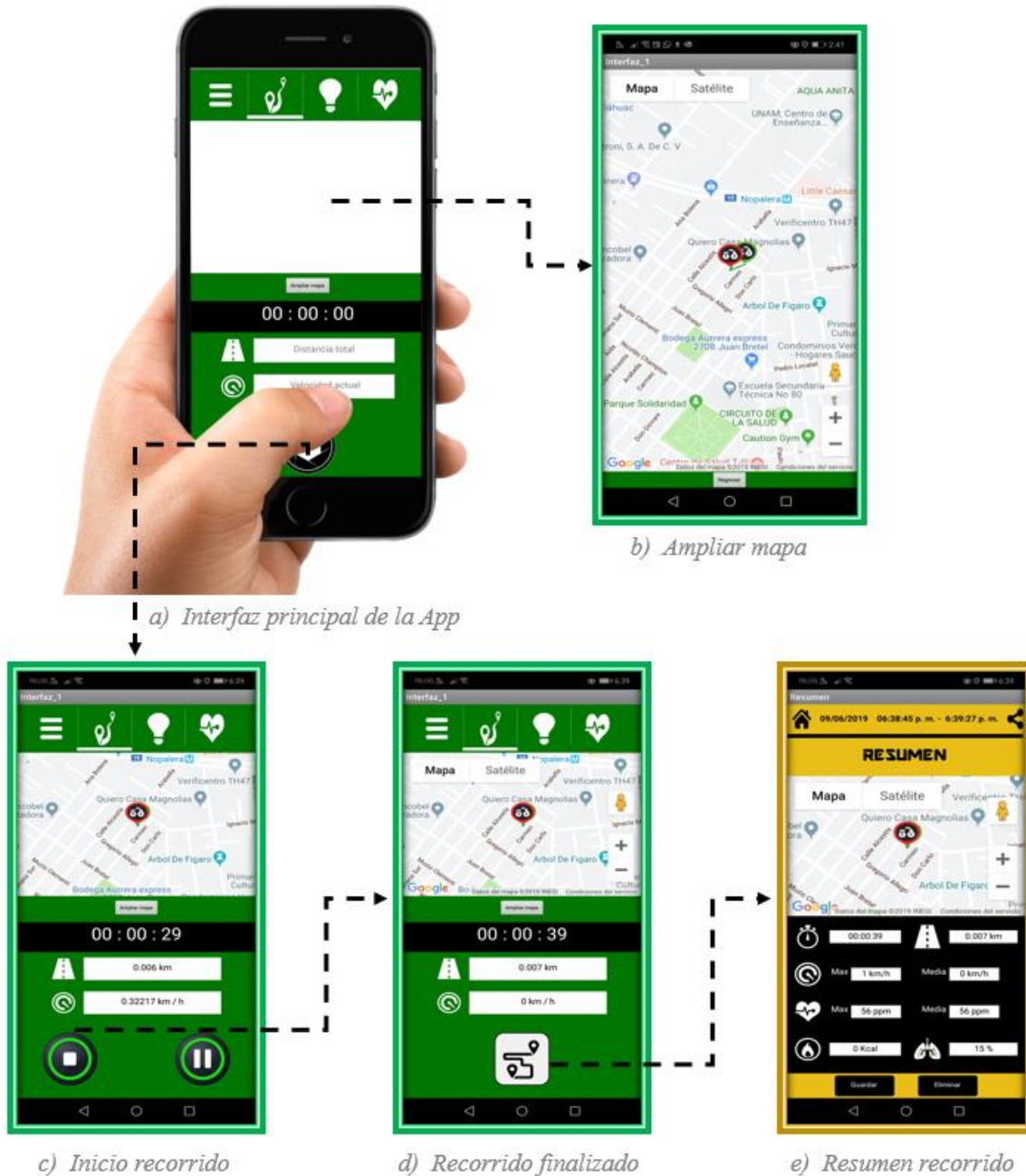


Figura 4. 7 Proceso de geolocalización de y resumen de recorrido

El cronómetro es una aplicación del reloj interno del Smartphone, la distancia es obtenida a través de la aplicación de la fórmula de Haversine valiéndose de los valores de coordenadas arrojados por el sensor de geolocalización del Smartphone, mientras que la velocidad promedio no es más que el cociente de las dos magnitudes anteriormente mencionadas.

La principal dificultad enfrentada en el desarrollo de esta sección tiene que ver con la aplicación del algoritmo de Haversine, sobre todo por su inexactitud de sus resultados en determinadas circunstancias, con lo que el cálculo de los demás parámetros también se ve afectado. Por esta razón, se tuvieron que realizar algunas modificaciones al algoritmo aplicado inicialmente con el objetivo de obtener resultados más apegados con las situaciones en que fue valorada la app, consiguiendo una tolerancia con márgenes aceptables.

Por otra parte también se cuenta con los botones de inició, pausa, finalización y resumen del recorrido para que el usuario pueda disponer de ellos de acuerdo a sus necesidades, así como ampliar el mapa para poder observarlo más detalladamente a pantalla completa.

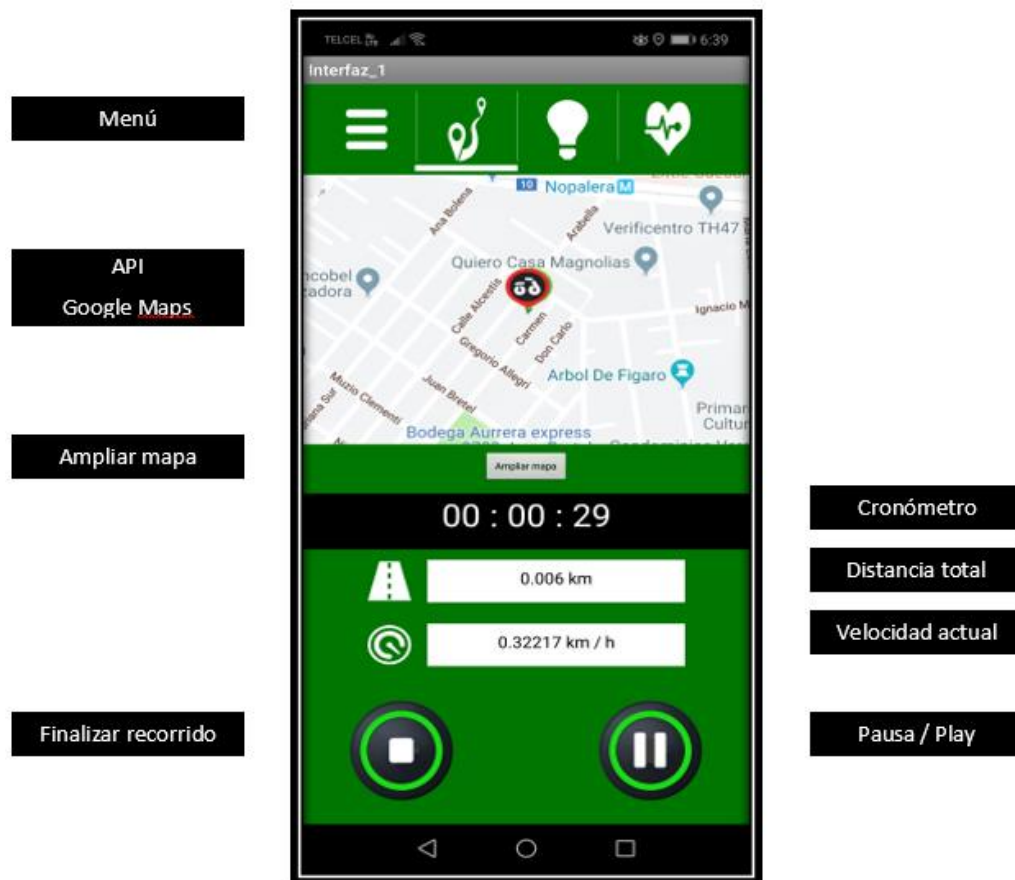


Figura 4. 8 Componentes de la interfaz "Geolocalización"

El mapa de geolocalización consiste en un visor web vinculado con la página web de Google Maps, las coordenadas de ubicación se obtienen a partir del sensor de geolocalización instalado en el Smartphone, con lo que es posible dar seguimiento al usuario. Es posible ampliar el tamaño del mapa presionando sobre el botón “Ver mapa” como se puede observar en la *figura 4.8*

El principal inconveniente presentado en este punto, y de la app en general, fue la implementación de la API de Google Maps relativas a la identificación exacta de las coordenadas del punto de ubicación dentro del mapa, así como el seguimiento de la ruta del usuario.

Tomando en consideración que los primeros vínculos establecidos con la aplicación de Google Maps eran incapaces de actualizarse en el transcurso de tiempo, fue necesaria la modificación y optimización del software, con lo cual se hizo posible rastrear la ubicación cada intervalo de tiempo determinado. Para ello fue necesario hacer uso de un archivo .html y vincularla con la app.

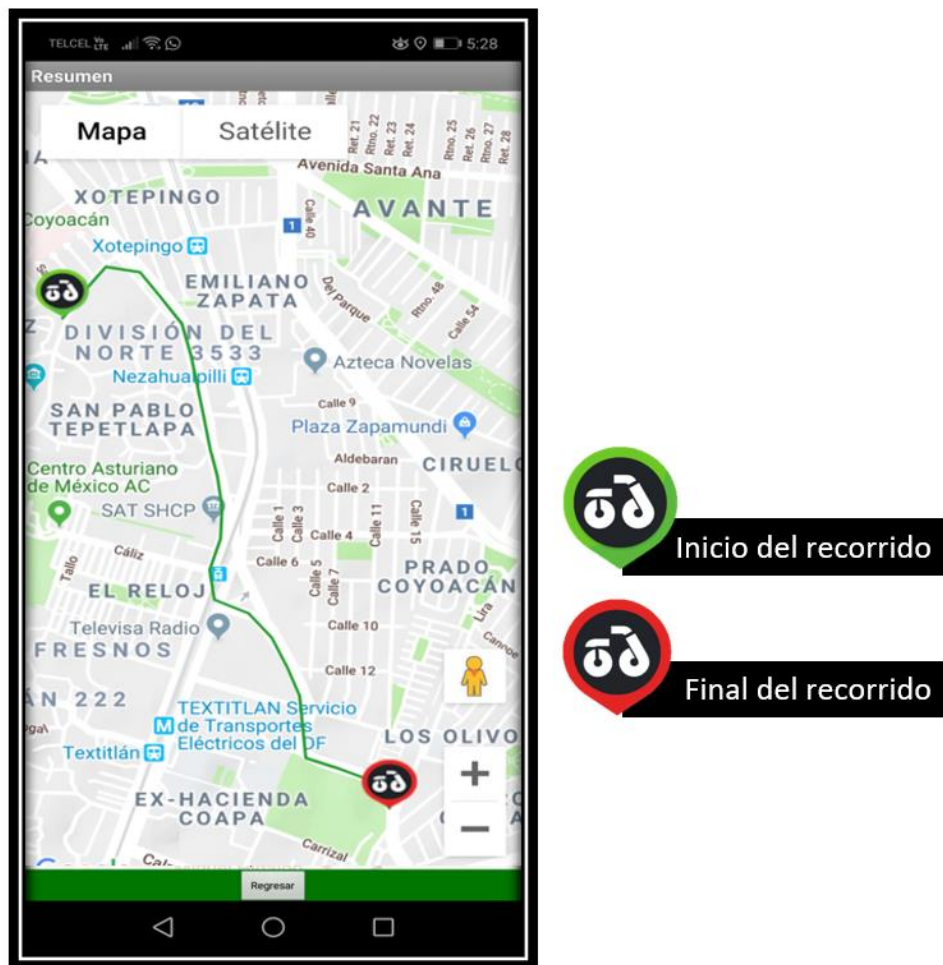


Figura 4. 9 Componentes de la interfaz "Google Maps"

Una vez que el usuario ha finalizado su recorrido, aparece un botón en la interfaz principal que le permite disponer de un análisis general del recorrido, así como visualizar su ruta en el mapa. Los parámetros con que cuenta el resumen del recorrido son: Tiempo, distancia, Velocidad máxima y promedio, Frecuencia máxima y promedio, Total de calorías quemadas y el promedio de Carga aeróbica durante su recorrido, además de la fecha, hora de inicio y hora de término de recorrido.

Paralelamente es posible para el usuario compartir los detalles de su recorrido en sus Redes Sociales por medio de un botón, con lo cual es posible interactuar con la comunidad de ciclistas. Adicionalmente, el usuario tiene la opción de guardar el resumen de su recorrido o en su historial. O en su defecto, eliminar ese registro.

En esta interfaz se planea en un futura la elaboración de gráficas comparativas que ayuden al usuario a medir su desempeño, esto como punto de mejora



Figura 4. 10 Componentes de la interfaz "Resumen de recorrido"

Así mismo, en la parte superior de la interfaz se hace presente el menú de opciones a base de botones identificados a partir de sus correspondientes logotipos (Inicio, mapa de geolocalización, control de luces y parámetros de rendimiento físico)

A partir de ese menú es posible dirigirse a las diferentes interfaces que constituyen la app y al submenú, desde el cual se puede acceder al historial del usuario, cerrar sesión o finalizar la app.

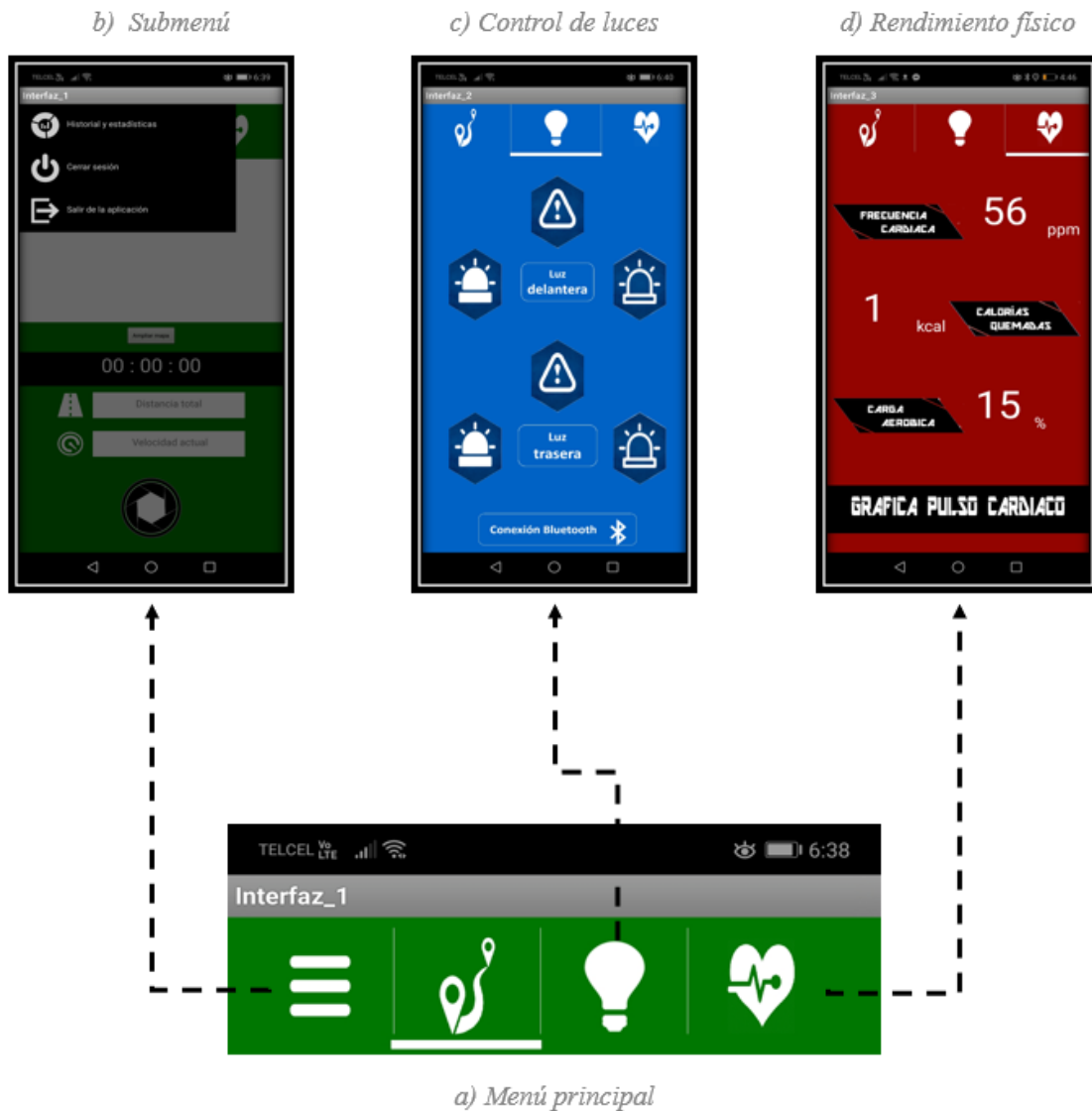


Figura 4. 11 Componentes del “Menú principal” de la app

4.1.5 Control de luces

La segunda opción del menú principal está representada por una serie de botones que se encargan de establecer la comunicación vía Bluetooth con las luces delantera y trasera para su control durante el recorrido, no sin antes realizar la correspondiente vinculación del dispositivo como se muestra en la *figura 4.12*.



Figura 4. 12 Procedimiento para el control de luces desde una app

La interfaz que se encarga del control de luces de la bicicleta está conformada por un botón de conexión, que se encarga de vincular el Smartphone con el módulo de iluminación por medio del módulo HC05 a través de su dirección MAC y el nombre del dispositivo previamente configurado, tal como se muestra en la *figura 4.12 (b)*

Una vez establecida la conexión bluetooth, es posible controlar el estado de las luces delantera y trasera, a través de tres botones correspondientes al encendido, apagado e intermitente de las luces, como es posible observar en la figura 4.13.

Todo este proceso está regulado a través de un sketch cargado al microcontrolador contenido en la plataforma Arduino Uno, quien se encarga de proveer las señales de control necesarias al módulo de relevadores, para llevar a cabo el encendido y apagado de las luces.

La corriente suministrada por la batería que alimenta a las luces es de particular importancia, fue ese impedimento el más importante presentado en esta sección, por lo que se tuvieron que corregir errores de lógica en el programa de Arduino, así como ajustar la pila de alimentación del sistema para conseguir un buen funcionamiento del mismo.

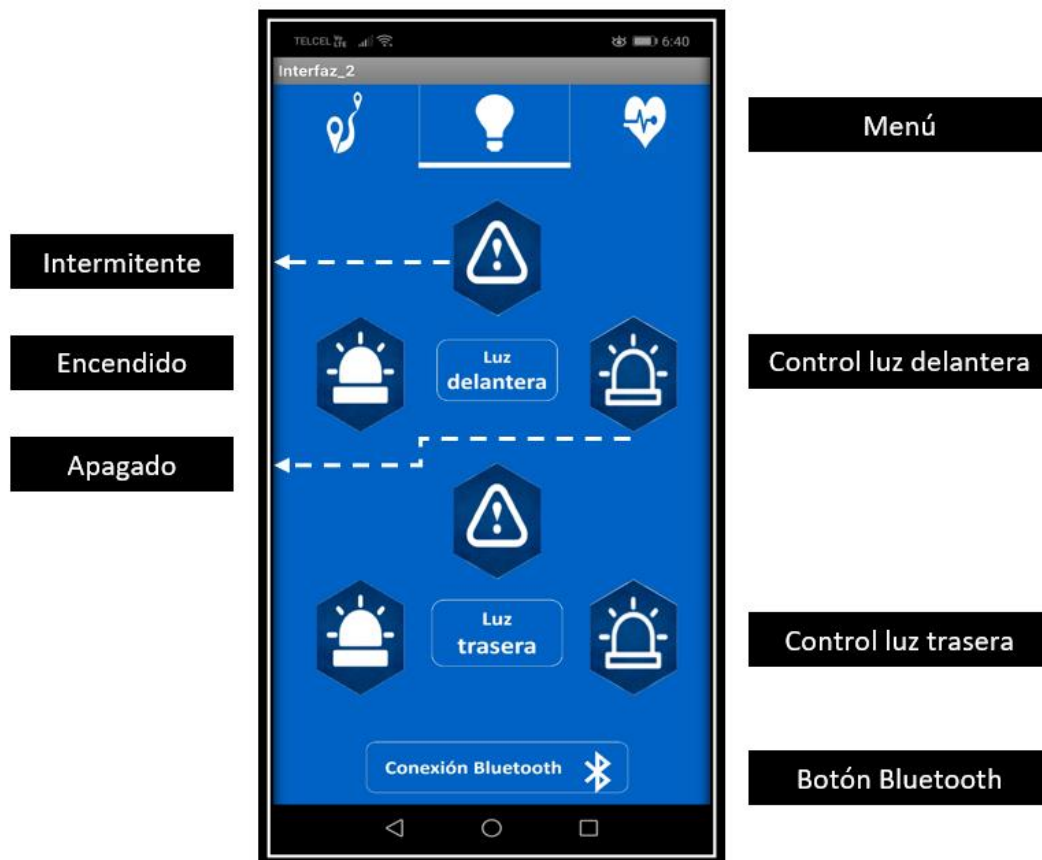


Figura 4.13 Componentes de la interfaz "Control de luces"

4.1.6 Parámetros de rendimiento físico y pulso cardiaco

La última opción dentro del menú principal de la app viene dada por la estimación de tres grandes parámetros de rendimiento físico: frecuencia cardíaca, calorías quemadas y carga aeróbica; que ayudan al usuario a hacer un análisis de su rendimiento físico durante el recorrido, tal como se muestra en la *figura 4.14*

Dichos parámetros actualizan sus valores cada cierto tiempo, el sistema se encarga de hacer las estimaciones y cálculos necesarios, mientras que el usuario se encuentra en pleno recorrido.

Es necesario considerar que los algoritmos empleados para el cálculo de los parámetros anteriormente mencionados son estimaciones hechas a partir de las condiciones iniciales presentadas por el usuario y el tipo de recorrido realizado.

Por ello se presentaron ciertas dificultades, que conllevaron diversas modificaciones sobre el programa inicial, para determinar un diagnóstico inicial del usuario y ajustar las variables que intervienen durante el recorrido con el objetivo de proporcionar al usuario parámetros dentro de un margen de tolerancia no tan distante de los valores reales o exactos, aspecto que podría ser mejorado a través del uso de aparatos de medición portátiles.

Cabe mencionar, que cada vez que el usuario realiza un cambio de interfaz a través del menú, las demás interfaces siguen trabajando en sus respectivas funciones, paralelamente a la utilizada en ese momento.

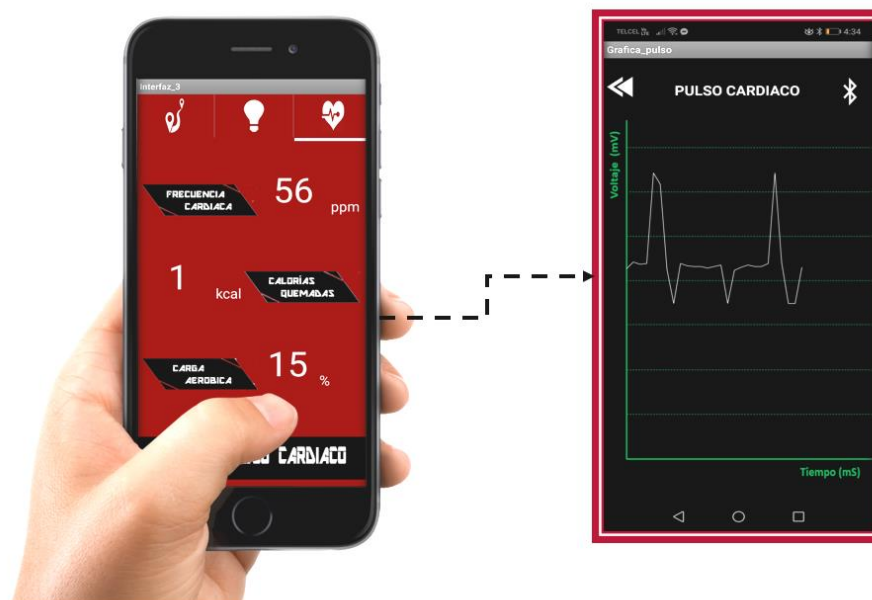


Figura 4.14 Proceso para despliegue de parámetros de rendimiento y grafica de pulso

Otra parte importante del proyecto tiene que ver con la grafica de pulso cardiaco, la cual se ejecuta en tiempo real, para lo cual es necesario realizar la comunicación entre el sensor de pulso cardiaco, el cual aprovecha la diferente luminosidad presentada al generarse un pulso cardiaco, y la app.

De esta forma se configura el sketch correspondiente en Arduino para que este a su vez pueda interpretar la información que llega de parte del sensor y enviarla a la app para su tratamiento.

La app hace un muestre de información cada 1 milisegundos con el objetivo de poder construir la grafica correspondiente, a través de un lienzo.

El sensor de pulso cardiaco esta diseñado para emplearse en cualquier dedo de la mano y cada vez que se genera un pulso cardiaco se enciende un led verde, con lo cual el usuario gráficamente pude verificar su comportamiento cardíaco mientras hace empleo de la bicicleta.

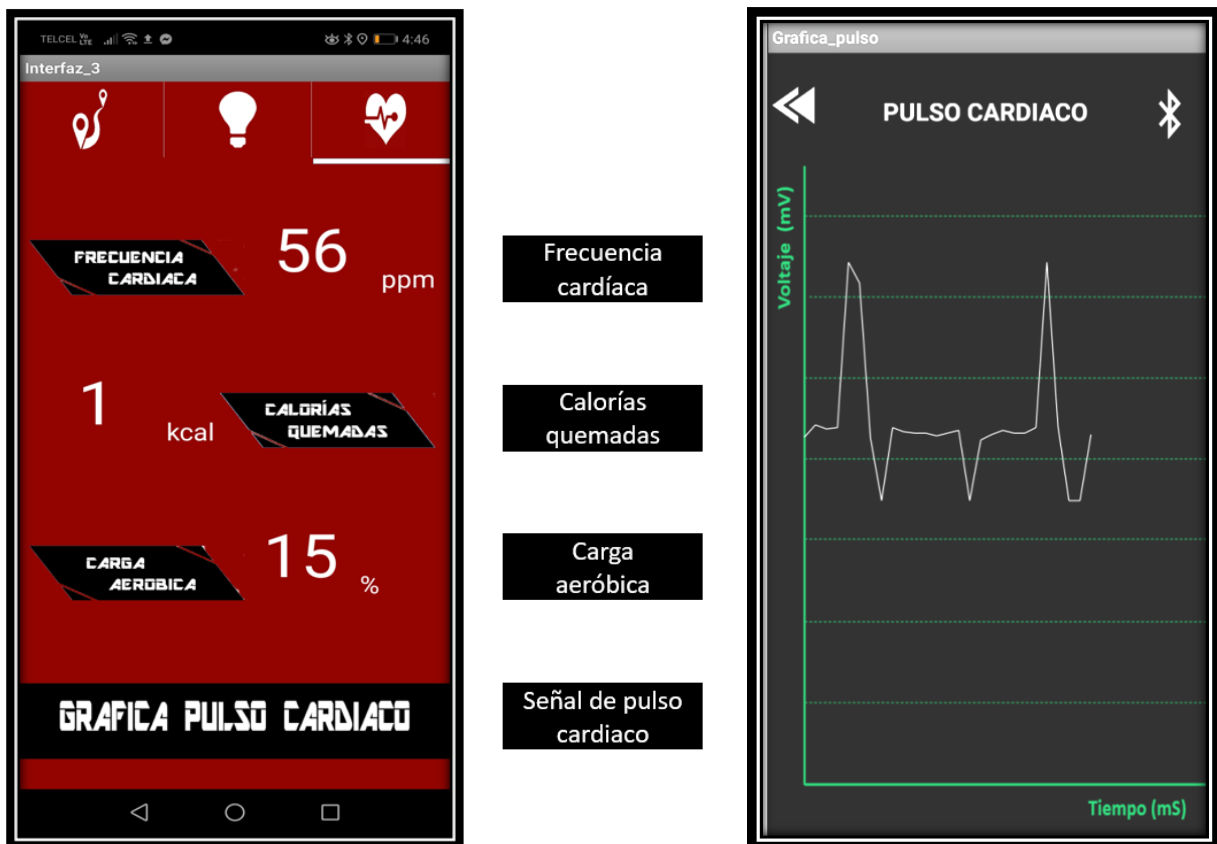


Figura 4. 15 Componentes de la interfaz "Parámetros de rendimiento físico"

4.1.7 Interfaces que componen la App



Figura 4. 16 Interfaces que componen la App "Eco Bici"

4.2 PRUEBAS E INSTALACIÓN DEL SISTEMA

Para complementar el funcionamiento de la app, se procede a instalar el sistema previamente diseñado y fabricado, el cual consiste básicamente en:

- ❖ Un módulo que integra la plataforma Arduino UNO, el módulo Bluetooth HC-05 y un conjunto de relevadores, así como el sensor de pulso cardíaco, la batería de 9 volts y un regulador de voltaje
- ❖ Una lámpara de leds de luz blanca y una tira de leds color rojo, que constituyen la iluminación frontal y trasera respectivamente.
- ❖ Un sujetador para Smartphone especial para bicicletas con la finalidad de proteger el dispositivo y poder hacer uso de este durante el recorrido.



Figura 4.17 Módulo de iluminación



Figura 4.18 Luces trasera y delantera



Figura 4.19 Sujetador para Smartphone

Una vez teniendo todos los elementos del sistema se realizan las pruebas correspondientes, con el problema de que la capacidad de corriente suministrada por la batería inicial es insuficiente para alimentar a nuestro circuito, por lo que se necesario recurrir a otra con características más adecuadas para nuestro proyecto.

Posteriormente, se detecta una falla de lógica dentro del programa alojado en la plataforma Arduino, causante de una incongruencia en la app, por lo que se hace necesario reescribir las instrucciones correspondientes. Las figuras 4.20 y 4.21 muestran los componentes que conforman el sistema de iluminación durante el período de prueba de este, incluyendo la app correspondiente al encendido de luces:



Figura 4.20 Prueba de luz delantera y trasera



Figura 4.21 Prueba de la app

De la misma forma, se realizan pruebas para corroborar el correcto funcionamiento del sensor de pulso cardíaco. Encontrándose con el problema de que las mediciones arrojadas por el sensor se tornan demasiado inestables, dificultando la construcción de la gráfica de pulso cardíaco. Por ello, se hace necesario la construcción de un adaptador que permita estabilizar la posición del sensor con respecto al dedo del usuario y así obtener mediciones más certeras. Las siguientes figuras 4.22 y 4.23 muestran el proceso de prueba del sensor de pulso cardíaco:

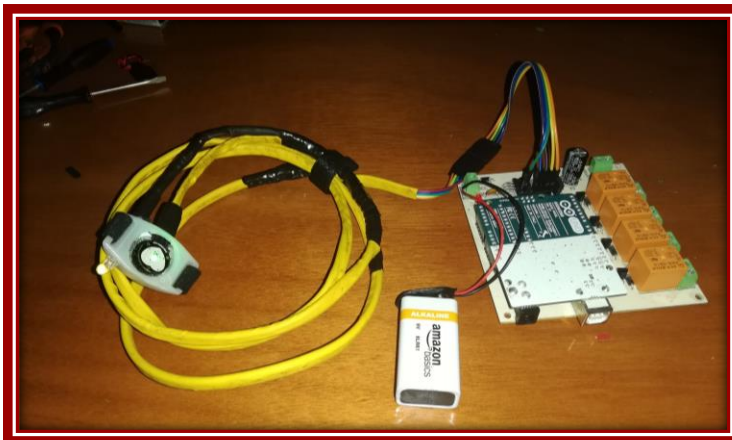


Figura 4.22 Prueba de sensor de pulso cardíaco



Figura 4.23 Adaptador de sensor

La siguiente *figura 4.24* muestra la distribución de los componentes ya sobre la bicicleta:

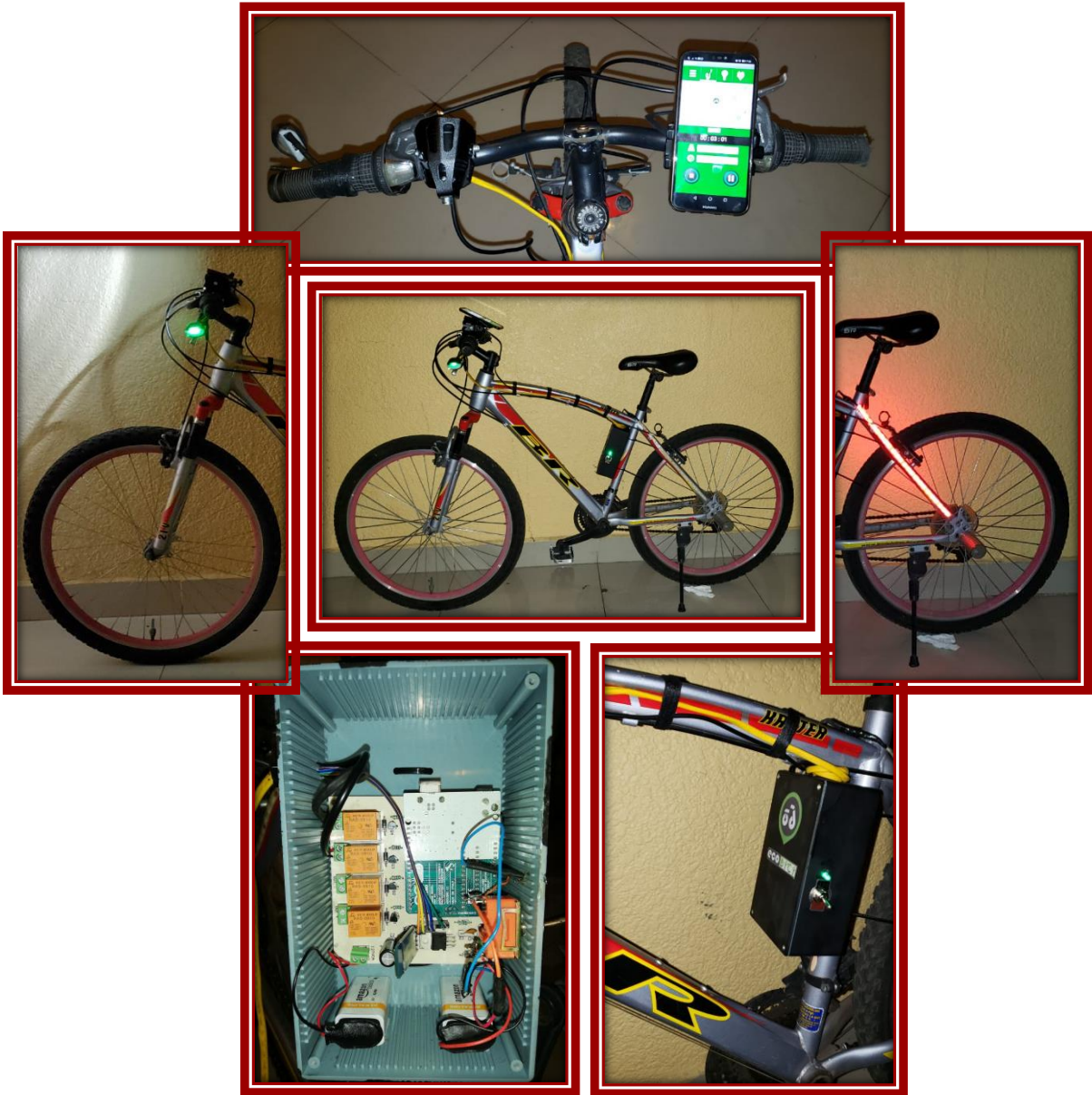


Figura 4.24 Instalación de sistema para bicicleta



CONCLUSIONES

Eco Bici es un proyecto que enfoca conocimientos de ingeniería en la elaboración de una app, que no sólo proporciona al usuario una práctica herramienta para la estimación de su rendimiento físico y optimiza el uso de la bicicleta, sino que también coadyuva al desarrollo sustentable de la vida cotidiana, en un mundo donde la tecnología cada día toma más relevancia.

Al día de hoy, los avances que se tienen respecto al proyecto son los siguientes:

Desarrollo de una app compatible con el sistema operativo Android a través de la plataforma MIT App Inventor 2, compuesta de tres interfaces principales enfocadas en la geolocalización, iluminación y medición de rendimiento físico del usuario durante su recorrido en bicicleta. Dicho desarrollo, lleva implícita la compilación e instalación de la App dentro del sistema operativo de un Smartphone.

Es necesario tomar en cuenta, que al no ser propiamente MIT App Inventor 2 un entorno de desarrollo integrado basado en algún lenguaje de programación no es apta para apps con funciones de alto grado de complejidad y no ofrece mucha flexibilidad sobre la modificación de proyectos, además de que sólo es compatible con el sistema operativo Android.

Otro de los puntos completados del proyecto es el vínculo establecido entre la app y Google Maps con el objetivo de mostrar la geolocalización del usuario, obteniendo también el tiempo de recorrido, la distancia y la velocidad aprovechando las coordenadas geográficas emitidas por los equipos Smartphone en tiempo real.

Cabe mencionar que, al hacer uso de la aplicación de Google Maps como parte fundamental, el funcionamiento óptimo de la app en general está a expensas de la calidad de conexión del dispositivo Smartphone del usuario a alguna red inalámbrica WiFi. Por lo que las fallas en dicha conexión de cualquier índole afectan directamente sobre la información procesada y mostrada en la App.

Así mismo, el proceso de implementación del módulo de iluminación sobre la bicicleta (Batería, Arduino 1, módulo HC05, Módulo de relevadores), así como una luz frontal y otra posterior; cuyos estados (encendido, apagado e intermitente) son controlados a través de botones contenidos en la app por medio de Bluetooth, ha sido concretado de manera exitosa.

En este punto, es importante considerar que para funcionamiento del módulo de iluminación se hace uso de una batería, cuya vida útil tiene una capacidad, por lo



que se hace necesario el cambio o carga de la misma, dependiendo la frecuencia de uso, para poder proporcionarles a los demás la energía suficiente para funcionar.

De igual forma, se ha logrado el desarrollo de una base de datos, encargada de recopilar y almacenar información específica de los usuarios, para su posterior uso en ciertos algoritmos aplicables para la estimación, a través de la app, de los parámetros de rendimiento físico más representativos para el usuario durante su recorrido (Frecuencia cardíaca, calorías quemadas y carga aeróbica)

Si bien, las mediciones obtenidas no cumplen las rigurosas normas establecidas para dispositivos médicos de alta exactitud, es posible realizar una evaluación objetiva del rendimiento físico mediante las estimaciones proporcionadas a través de la app, ya que los resultados, si bien no son del todo exactos, son consistentes, lo cual resulta útil al momento de establecer marcas personales y controlar nuestro ritmo de trabajo.

Como todo proyecto, el presente trabajo no estuvo exento de problemas técnicos y errores humanos, con los que se tuvo que lidiar, habiéndose enriquecido así la capacidad de solución de problemas y trabajo en equipo. Tras ciertas situaciones experimentales y nuevos conocimientos adquiridos por medio de las diversas fuentes de información consultadas, se ha llegado a dar al proyecto un enfoque diferente a la inicial.

Los cambios que se suscitaron en los objetivos iniciales están enfocados en hacer de la app algo más práctico y más funcional, para así poder ofrecer una herramienta más completa y fiable al usuario.

Los resultados obtenidos a lo largo del proyecto reflejan que, si bien, es necesario detallar ciertos aspectos en la app para hacer de ella algo funcional en la vida cotidiana de los usuarios, es posible su futura inserción dentro del mercado de las aplicaciones, ya que se trata de un concepto útil para la sociedad e innovador.

De esta forma, existe la posibilidad de anexar nuevas y más complejas funciones al proyecto en miras de presentar un producto más útil a la sociedad, tales como: Un cargador para dispositivos móviles a partir de un dinamo, agregar al módulo de iluminación un sensor de cercanía de objetos para alarmar al usuario, así como un sistema de luces direccionales, un mando o manillar para controlar las funciones de la app de manera más práctica y hasta un sistema antirrobo vinculado con el smartphone del usuario.



ANEXOS



ATmega328P

8-bit AVR Microcontroller with 32K Bytes In-System
Programmable Flash

DATASHEET

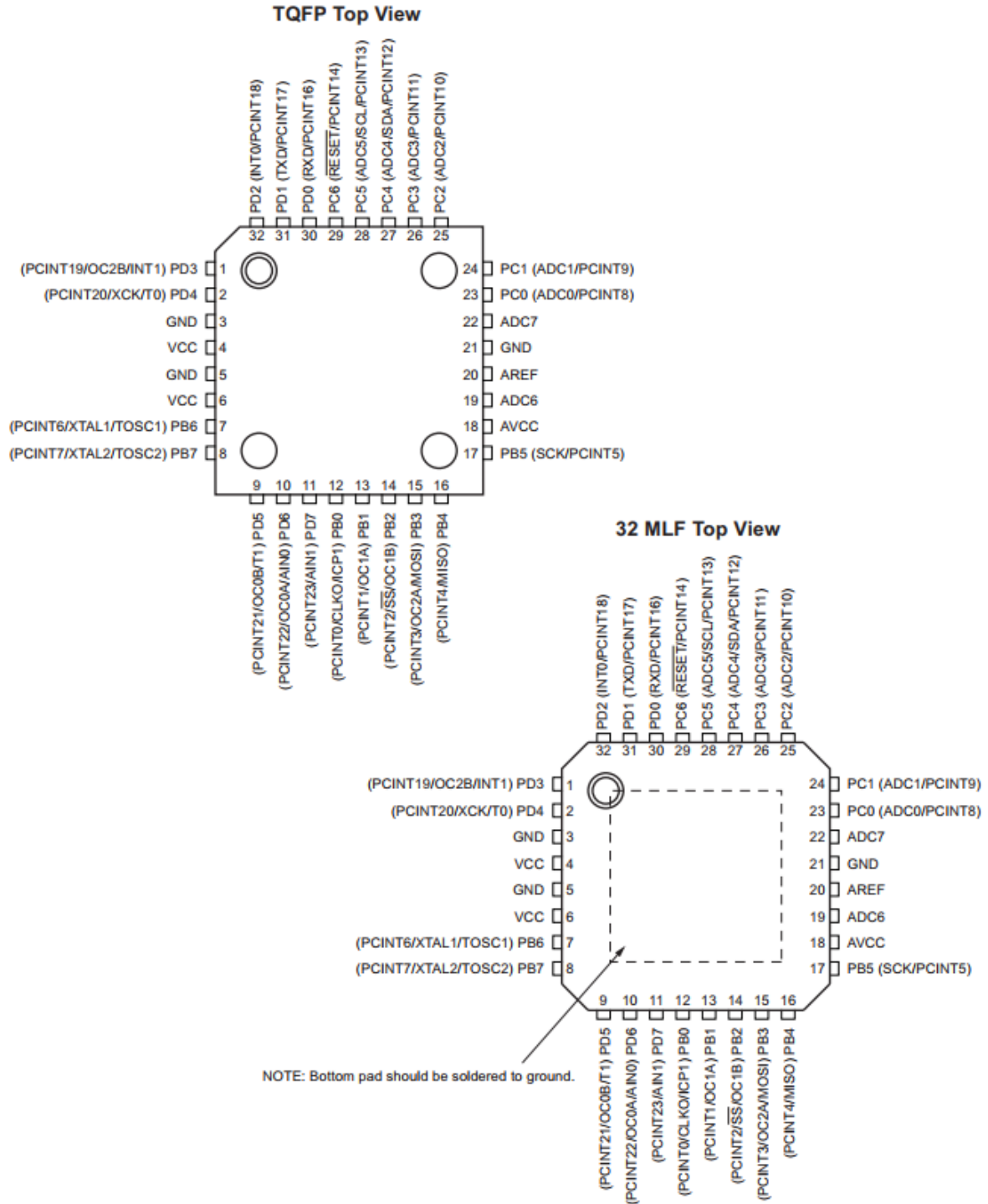
Features

- High performance, low power AVR® 8-bit microcontroller
- Advanced RISC architecture
 - 131 powerful instructions – most single clock cycle execution
 - 32 × 8 general purpose working registers
 - Fully static operation
 - Up to 16MIPS throughput at 16MHz
 - On-chip 2-cycle multiplier
- High endurance non-volatile memory segments
 - 32K bytes of in-system self-programmable flash program memory
 - 1Kbytes EEPROM
 - 2Kbytes internal SRAM
 - Write/erase cycles: 10,000 flash/100,000 EEPROM
 - Optional boot code section with independent lock bits
 - In-system programming by on-chip boot program
 - True read-while-write operation
 - Programming lock for software security
- Peripheral features
 - Two 8-bit Timer/Counters with separate prescaler and compare mode
 - One 16-bit Timer/Counter with separate prescaler, compare mode, and capture mode
 - Real time counter with separate oscillator
 - Six PWM channels
 - 8-channel 10-bit ADC in TQFP and QFN/MLF package
 - Temperature measurement
 - Programmable serial USART
 - Master/slave SPI serial interface
 - Byte-oriented 2-wire serial interface (Phillips I²C compatible)
 - Programmable watchdog timer with separate on-chip oscillator
 - On-chip analog comparator
 - Interrupt and wake-up on pin change
- Special microcontroller features
 - Power-on reset and programmable brown-out detection
 - Internal calibrated oscillator
 - External and internal interrupt sources
 - Six sleep modes: Idle, ADC noise reduction, power-save, power-down, standby, and extended standby



Pin Configurations

Figure 1-1. Pinout

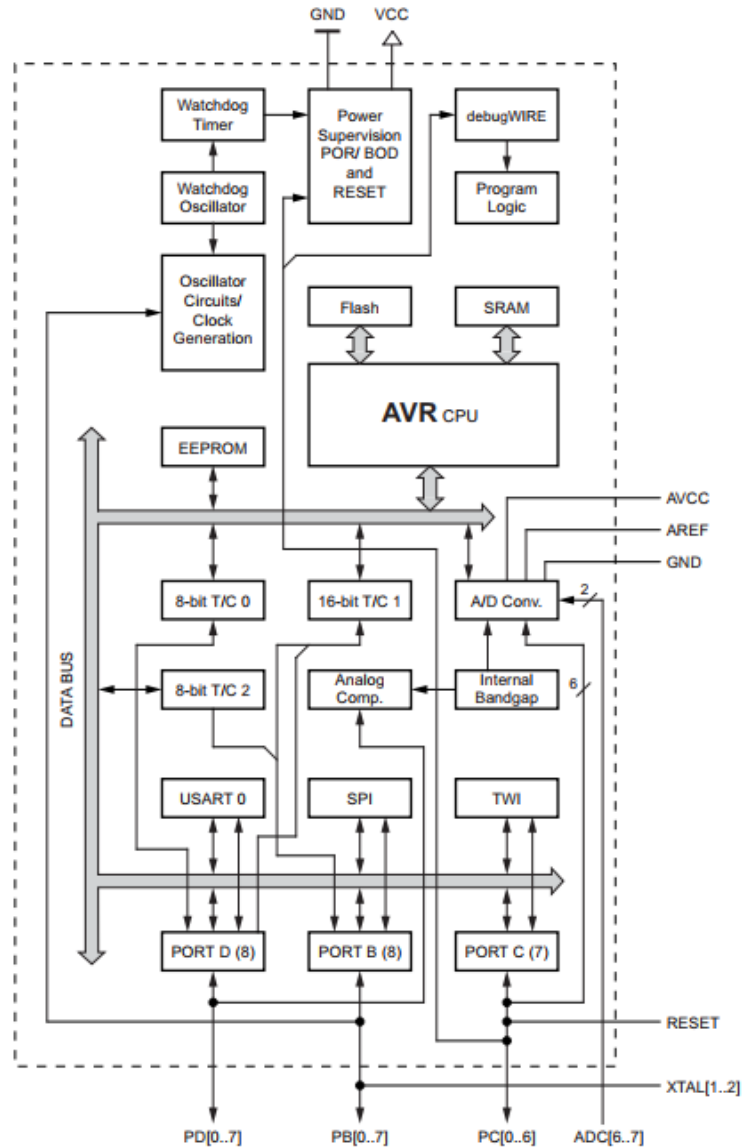


Overview

The Atmel® ATmega328P is a low-power CMOS 8-bit microcontroller based on the AVR® enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega328P achieves throughputs approaching 1MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

Block Diagram

Figure 2-1. Block Diagram





November 2014

1N4001 - 1N4007 General-Purpose Rectifiers

Features

- Low Forward Voltage Drop
- High Surge Current Capability



DO-41
COLOR BAND DENOTES CATHODE

Ordering Information

Part Number	Top Mark	Package	Packing Method
1N4001	1N4001	DO-204AL (DO-41)	Tape and Reel
1N4002	1N4002	DO-204AL (DO-41)	Tape and Reel
1N4003	1N4003	DO-204AL (DO-41)	Tape and Reel
1N4004	1N4004	DO-204AL (DO-41)	Tape and Reel
1N4005	1N4005	DO-204AL (DO-41)	Tape and Reel
1N4006	1N4006	DO-204AL (DO-41)	Tape and Reel
1N4007	1N4007	DO-204AL (DO-41)	Tape and Reel

Absolute Maximum Ratings

Stresses exceeding the absolute maximum ratings may damage the device. The device may not function or be operable above the recommended operating conditions and stressing the parts to these levels is not recommended. In addition, extended exposure to stresses above the recommended operating conditions may affect device reliability. The absolute maximum ratings are stress ratings only. Values are at $T_A = 25^\circ\text{C}$ unless otherwise noted.

Symbol	Parameter	Value							Unit
		1N 4001	1N 4002	1N 4003	1N 4004	1N 4005	1N 4006	1N 4007	
V_{RRM}	Peak Repetitive Reverse Voltage	50	100	200	400	600	800	1000	V
$I_{F(AV)}$	Average Rectified Forward Current .375 " Lead Length at $T_A = 75^\circ\text{C}$	1.0							A
I_{FSM}	Non-Repetitive Peak Forward Surge Current 8.3 ms Single Half-Sine-Wave	30							A
I^2t	Rating for Fusing ($t < 8.3$ ms)	3.7							A^2sec
T_{STG}	Storage Temperature Range	-55 to +175							$^\circ\text{C}$
T_J	Operating Junction Temperature	-55 to +175							$^\circ\text{C}$



Typical Performance Characteristics

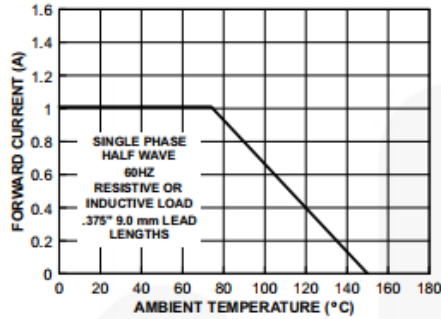


Figure 1. Forward Current Derating Curve

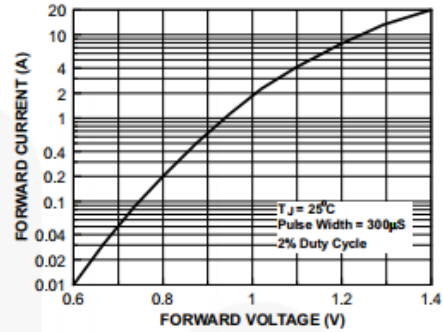


Figure 2. Forward Characteristics

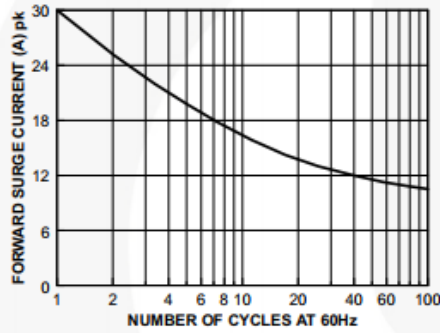


Figure 3. Non-Repetitive Surge Current

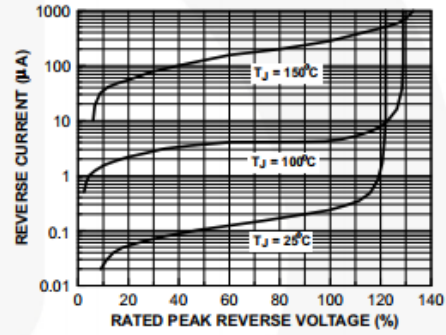


Figure 4. Reverse Characteristics



APDS-9008 Miniature Surface-Mount Ambient Light Photo Sensor



Data Sheet



Description

The APDS-9008 is a low cost analog-output ambient light photo sensor in miniature chipLED lead-free surface mount package. It consists of a spectrally suited photo sensor, which provides excellent responsivity that is close to the response of the human eyes, as show in figure 2. The APDS-9008 is ideal for applications in which the measurement of ambient light is used to control display backlighting. Mobile appliances such as the mobile phones and PDAs that draw heavy current from display backlighting will benefit from incorporating these photo sensor products in their designs by reducing power consumption significantly.

Application Support Information

The Application Engineering Group is available to assist you with the application design associated with APDS-9008 ambient light photo sensor module. You can contact them through your local sales representatives for additional details.

Features

- Excellent responsivity
 - Close responsivity to the human eye
- Miniature ChipLED Leadfree surface-mount package
 - Height – 0.55 mm
 - Width – 1.60 mm
 - Depth – 1.50 mm
- Low sensitivity variation across various light sources
- Operating temperature : -40°C to 85°C
- Vcc supply 1.6 to 5.5V
- Lead-free package, RoHS compliance
- Output linearity across wide illumination range
- High output saturation voltage

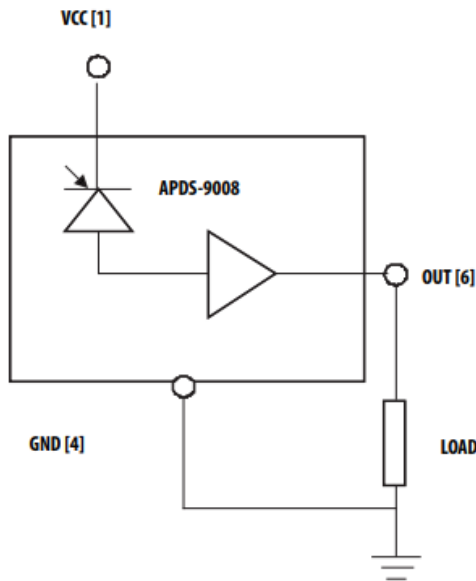
Applications

- Detection of ambient light to control display backlighting
 - Mobile devices – Mobile phones, PDAs
 - Computing devices – Notebooks, Webpads
 - Consumer devices – TVs, Video Cameras, Digital Still Camera
- Automatic Residential and Commercial Lighting Management
- Electronic Signs and Signals

Ordering Information

Part Number	Packaging Type	Package	Quantity
APDS-9008-020	Tape and Reel	6-pins Chiplid package	2500

Typical Application Circuit



I/O Pins Configuration Table

Pin	Symbol	Description
1	VCC	VCC
2	NC	No Connect
3	NC	No Connect
4	GND	Ground
5	NC	No Connect
6	lout	Out

Figure 1. Typical application circuit for APDS-9008

Absolute Maximum Ratings

For implementations where case to ambient thermal resistance is $\leq 50 \text{ }^\circ\text{C} / \text{W}$

Parameter	Symbol	Min.	Max.	Units
Storage Temperature	T_S	-40	85	$^\circ\text{C}$
Supply Voltage	V_{CC}	0	6	V

Recommended Operating Conditions

Parameter	Symbol	Min.	Max.	Units	Conditions
Operating Temperature	T_A	-40	85	$^\circ\text{C}$	
Supply Voltage	V_{CC}	1.6	5.5	V	



GLOSARIO



ANDROID: Es un sistema operativo inicialmente pensado para teléfonos móviles. Está basado en Linux, tiene un núcleo de sistema operativo libre, es gratuito y multiplataforma

APK: Android Application Package, Es el formato de archivo utilizado para la instalación de software en el sistema operativo Android, una archivo .apk contiene tanto la aplicación en sí misma como el instalador que permite que se pueda guardar y ejecutar en tu terminal

APP: Es una abreviatura de la palabra en inglés application. Es decir, es una aplicación informática diseñada para ser ejecutada en teléfonos inteligentes, tabletas y otros dispositivos móviles y que permite al usuario efectuar una tarea concreta de cualquier tipo

ATP: Trifosfato de adenosina, molécula que se encuentra en todos los seres vivos y constituye la fuente principal de energía utilizable por las células para realizar sus actividades.

BLUETOOTH: Es una especificación industrial para Redes Inalámbricas de Área Personal que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia en la banda ISM de los 2.4 GHz.

CAPTCHA: Completely Automated Public Turing test to tell Computers and Humans Apart, es un tipo de medida de seguridad que demuestra si quien intenta acceder a una cuenta protegida por contraseña es humano y no un ordenador.

CSS: Cascading Style Sheets, es un lenguaje que define la apariencia de un documento escrito en un lenguaje de marcado, por ejemplo, HTML.

GADGET: Es un dispositivo que tiene un propósito y una función específica, generalmente de pequeñas proporciones, práctico y a la vez novedoso.

GOOGLE MAPS: Es un servidor de aplicaciones de mapas en la web que pertenece a Alphabet, Inc. Ofrece imágenes de mapas desplazables, así como fotografías por satélite del mundo e incluso la ruta entre diferentes ubicaciones o imágenes a pie de calle.

GPS: Global Positioning System, es un sistema que permite determinar en toda la Tierra la posición de un objeto (una persona, un vehículo) con una precisión de hasta centímetros. El sistema fue desarrollado, instalado y empleado por el Departamento de Defensa de los Estados Unidos



HTML: HyperText Markup Language, es un estándar que sirve de referencia del software que conecta con la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código para la definición de contenido de una página web

IDE: Integrated Development Environment, es una aplicación informática que proporciona servicios integrales para facilitar al desarrollador o programador el desarrollo de software.

IOS: Es un sistema operativo móvil de la multinacional Apple Inc. Originalmente desarrollado para el iPhone (iPhone OS), después se ha usado en dispositivos como el iPod touch y el iPad. No permite la instalación de iOS en hardware de terceros

ISM: Industrial, Scientific and Medical, son bandas reservadas internacionalmente para uso no comercial de radiofrecuencia electromagnética en áreas industrial, científica y médica.

JAVASCRIPT: Es un lenguaje de guiones (*scripts*) interpretado (no compilado) que funciona como una extensión de HTML.

KAWA: Es un lenguaje de programación para la plataforma Java. Puede ser utilizado como un lenguaje interpretado, es decir, diseñado para ser ejecutado por medio de un intérprete, sin embargo, incluye un compilador.

LCD: Liquid Crystal Display, es una pantalla delgada y plana formada por un número de píxeles en color o monocromos colocados delante de una fuente de luz o reflectora.

LED: Ligth Emitting Diode, es una fuente de luz constituida por un material semiconductor dotado de dos terminales.

MAC: Media Access Control, Una dirección MAC es el identificador único asignado por el fabricante a una pieza de hardware de red

PWM: Pulse Width Modulation, es una técnica de modulación en la que se modifica el ciclo de trabajo de una señal periódica para transmitir información a través de un canal de comunicaciones.

SDK: Software Development Kit, es un conjunto de herramientas que ayudan a la programación de aplicaciones para un entorno tecnológico particular.

SKETCH: Parte de código fuente listo para abrir con el entorno de desarrollo integrado de Arduino y ser cargado sobre el microcontrolador de la plataforma.



SMARTPHONE: Es un término comercial para denominar a un teléfono móvil que incorpora una mayor capacidad de procesamiento de datos y conectividad a internet que los dispositivos tradicionales.

SPLASH SCREEN: Elemento visual de la aplicación, consiste en una imagen que se muestra en pantalla, normalmente centrada, mientras esperamos que la aplicación arranque.

SPP: *Serial Port Profile*, emula una línea serie y provee una interfaz de reemplazo de comunicaciones basadas en RS-232, con las señales de control típicas.

USB: *Universal Serial Bus*, es un tipo de dispositivo de almacenamiento de datos que utiliza memoria flash para guardar datos e información.

WPAN: *Wireless Personal Area Network*, incluye redes inalámbricas de corto alcance que abarcan un área de algunas decenas de metros. Se usa generalmente para conectar dispositivos periféricos a un ordenador sin conexión por cables



BIBLIOGRAFÍA



INTRODUCCIÓN

- [1] Servín, M.A., Gómez, L., y González, R. (14 de diciembre de 2015). *Insuficientes, planes de movilidad en el DF. LaJornada*, pp. 2.
- [2] Reporte Nacional de Movilidad Urbana 2014-2015, Por un mejor futuro urbano, pp. 35-38
- [3] Navarro, I. (20 de agosto del 2014). *Autos, primera fuente de contaminación en el país. Milenio*. Obtenido de: http://www.milenio.com/region/Autos-primera-fuente-contaminacion-pais-Mexico-enfermedades-humo-ozono_troposferico_0_338966122.html
- [4] Instituto Nacional de Estadística y Geografía (15 de noviembre del 2016). *Estadísticas a propósito del día mundial en recuerdo de las víctimas de los accidentes de tránsito*
- [5] Encuesta Nacional de Nutrición y Salud 2012, *Actividad física*, pp. 86
- [6] Instituto para el Transporte y el Desarrollo de Políticas (2016). *Estimación de beneficios ambientales del Desarrollo Orientado al Transporte*. Obtenido de: <http://itdp.mx/dotmx/#/estimaciones>

ESTADO DEL ARTE

- [7] BiciHome (2016). *Historia de la bicicleta*. Obtenido de: <http://bicihome.com/la-historia-de-las-bicicleta/>
- [8] Milenio Digital (2016). *Ocho apps que debes tener si andas en bici*. Obtenido de: http://www.milenio.com/tendencias/dia_internacional_bicicleta-apps_ciclistas-apps_bicicleta-ecobici-wikicleta_0_721728098.html
- [9] Ecoticias (2016). *Biciwood, una bicicleta inteligente y muy ecológica*. Obtenido de: <http://www.ecoticias.com/tecnologia-verde/166169/biciwood-una-bicicleta-inteligente-muy-ecologica>
- [10] Energética futura (agosto 2016). *Hammerhead, el navegador para bicicletas*. <https://energeticafutura.com/blog/hammerhead-el-navegador-para-bicicletas/>
- [11] Xataca (2016). *DuBike: La bicicleta inteligente de Baidu*. Obtenido de: <http://www.neoteo.com/dubike-la-bicicleta-inteligente-de-baidu>



[12] COBI (2016). *Descubre COBI: el primer sistema de bicicleta con conexión inteligente*.

Obtenido de:

<https://cobi.bike/es/producto>

[13] *Smarthalo* (2016). *Smarthalo, tu bici inteligente*. Obtenido de:

<https://www.smarthalo.bike/es/>

MARCO TEÓRICO

[14] Android Developers (2016). *Sensors overview*. Obtenido de:

http://developer.android.com/intl/es/guide/topics/sensors/sensors_overview.html

[15] Madrid, C. A. (12 de Mayo de 2016). *Aplicaciones Móviles*. Obtenido de:

<https://appsmovilescavucm.wordpress.com>

[16] Cuello, J. V. (2013). *Diseñando apps para móviles*. Obtenido de:

<http://appdesignbook.com/es/contenidos/las-aplicaciones/>

[17] App Inventor en español (2016). *Primeros pasos*. Obtenido de:

<https://sites.google.com/site/appinventormegusta/primeros-pasos>

[18] Torrente, A. O. (2013). *Arduino, curso práctico de formación*. Ciudad de México:

Alfaomega

[19] Arduino (2016). *¿Qué es Arduino?*. Obtenido de: <http://arduino.cl/que-es-arduino/>

[20] Neoteo. (2016). *Módulo Bluetooth HC-06*. Obtenido de:

<http://www.neoteo.com/modulo-bluetooth-hc-06-android>

[21] Led Resource. (2016). *Led flashlight review*. Obtenido de: <http://www.led-resource.com>

[22] Carreón, G.A., Martínez, M. A., y Treviño, T. X. (2011). *Manual del Ciclista Urbano de la Ciudad de México*. Ciudad de México: Designio editores

[23] Luces para bicicletas (2016). *Luces LED*. Obtenido de:

<http://www.lucesparabicicletas.com>

[24] Beachle, T., y Earle, R. (2000). *Principios del entrenamiento de fuerza y el acondicionamiento físico*. Madrid, España: Editorial Medica Panamericana



DISEÑO E IMPLEMENTACIÓN

[25] MrReid.org (12 de Diciembre de 2011). Finding the distance between two points on the Earth's surface. Obtenido de:

<http://wordpress.mrreid.org/2011/12/20/haversine-formula/>

[26] Merí, A. (2005) Fundamentos de fisiología de la actividad física y el deporte. Buenos Aires, Argentina: Panamericana, pp. 48

[27] Argudo, F., Ibañez, S. y Alonso, J. (2010) Softwares aplicados al entrenamiento e investigación en el deporte. Sevilla, España: Wenceulen Editorial Deportiva, pp. 80