



UNIVERSIDAD DEL BÍO-BÍO  
FACULTAD DE CIENCIAS EMPRESARIALES  
DEPARTAMENTO DE SISTEMAS DE INFORMACIÓN

---

**Desarrollo de Sistema Cyber Físico para monitoreo en tiempo real de ingreso y salida de vehículos en la Universidad del Bío-Bío, Campus Concepción.**

**Memoria para optar al Título de Ingeniero Civil en Informática**

**Autor**

Rodrigo Oliva Troncoso

**Profesor Guía**

Patricio Galdames Sepúlveda

---

Concepción, marzo 2019

## **Agradecimientos**

Ha sido un largo y duro camino, no puedo dejar fuera de este trabajo a tanta gente que me brindó su apoyo y ayuda en diversos momentos. Primero que todo, agradecido de Dios por estar en todo momento a mi lado; *Se fuerte y valiente! No tengas miedo ni te desanimas, porque el Señor tu Dios está contigo donde quiera que vayas. Josué 1:9.*

De igual manera a mi familia, por el apoyo incondicional a pesar de las dificultades que se presentaron a lo largo de estos años de estudio.

Por otro lado, quiero destacar el apoyo del Sr. Marco Iturra, quien me enseñó valiosas herramientas tanto como persona y de manera profesional, las cuales me hacen sentir seguro para enfrentar esta segunda parte que hoy vivo con mucho entusiasmo.

Finalmente, agradecer a muchos profesores y funcionarios que me dieron la posibilidad de poder desempeñarme en el ámbito laboral y social a lo largo de estos años. Simplemente gracias Programa de Tutores, Departamento de Matemática, Promoción de Carreras, Staff de carreras FACE, Departamento de Sistemas de Información, Laboratorio de Especialidad, Facultad de Ciencias Empresariales y en general a mi Universidad del Bío-Bío. Sin duda ha sido una gran experiencia, de mucho aprendizaje y me llevo los mejores recuerdos.

A cada uno de ellos les mando un abrazo de todo corazón, siento que este trabajo es reflejo de todo el apoyo y cariño a través de los años.

## Resumen

Este proyecto se desarrolla para dar conformidad a los requisitos exigidos por la Universidad del Bío-Bío en el proceso de titulación de la carrera de Ingeniería Civil en Informática de la Facultad de Ciencias Empresariales.

Hoy en día, la Universidad del Bío-Bío en su campus Concepción se ha visto reflejado su crecimiento, tanto en nuevos estudiantes como infraestructura, esto conlleva al aumento del flujo vehicular que hoy presenta la universidad. Actualmente no existe ningún tipo de registro ni estadística del flujo vehicular dentro de los distintos ingresos a estacionamientos emplazados en el campus Concepción de la Universidad del Bío-Bío. Generando el no poder hacer planes de contingencia ni obtener estadísticas de su uso, sin poder optimizar su capacidad de ninguna forma, lo cual hoy en día se ve reflejado en el gran tráfico vehicular que se ve afectado el campus, no existiendo un orden y control al momento del ingreso y salida de vehículos.

El siguiente proyecto consistió en el diseño y construcción de un Sistema Cyber-Físico de monitoreo en tiempo real del ingreso y salida de vehículo, se desarrolla por medio de un sistema físico, cumpliendo la función de integrar microcontroladores a una placa de circuito impreso, permitiendo por medio de sensores de distancia el control del flujo vehicular, este sistema se complementa con un sitio web, para la visualización de gráficos y el historial de la cantidad de vehículos de acuerdo al filtro seleccionado. Gracias a este sistema se pueden realizar informes que ayuden a la toma de decisiones y el mejorar el actual servicio de barreras.

## **Abstract**

This project is being developed to meet the requirements demanded by the University of Bio-Bio in the process of Civil Engineering degree in College of Business Administration.

Today, the Universidad del Bío-Bío on its Concepción campus has seen its growth reflected in both new students and infrastructure, this leads to an increase in the flow of vehicles that today presents the university. Currently there is no record or statistics of the flow of vehicles within the various parking incomes located on the campus Concepción of the Universidad del Bío-Bío. This is the not being able to make contingency plans or obtain statistics of its use, without being able to optimize its capacity in any way, which nowadays is reflected in the great vehicular traffic that is affected the campus, not existing an order and control at the moment of the entrance and exit of vehicles.

The present project consisted in the design and construction of a Cyber-Physical System of monitoring in real time of the entrance and exit of vehicles. It is developed by means of a physical system, fulfilling the function of integrating microcontrollers to a printed circuit board, allowing by means of sensors of distance the control of the vehicular flow, this system is complemented with a Web site, for the visualization of graphs and the history of the quantity vehicles according to the selected filter. Thanks to this system reports can be made to help decision making and improve the current service barriers.

## Índice general

<b>CAPÍTULO I INTRODUCCIÓN</b> .....	9
<b>1.1. Introducción</b> .....	10
<b>CAPITULO II DEFINICIÓN DE LA INSTITUCIÓN</b> .....	11
<b>2. Definición de la institución</b> .....	12
<b>2.1. Descripción de la institución</b> .....	12
<b>2.2. Descripción del área de estudio</b> .....	13
<b>2.3. Descripción de la problemática</b> .....	13
<b>CAPITULO III DEFINICIÓN DEL PROYECTO</b> .....	14
<b>3. Definición del proyecto</b> .....	15
<b>3.1. Objetivo general del proyecto</b> .....	15
<b>3.2. Objetivos específicos del proyecto</b> .....	15
<b>3.3. Lenguaje, herramientas y recursos</b> .....	16
<b>3.3.1. Ambiente de Ingeniería de Software</b> .....	16
<b>3.3.1.1. Metodología</b> .....	16
<b>3.3.1.2. Tecnologías a utilizar</b> .....	17
<b>3.3.1.3. Técnicas y notaciones</b> .....	17
<b>3.3.1.4. Estándares de documentación</b> .....	18
<b>3.3.1.5. Herramientas de apoyo</b> .....	18
<b>3.3.1.6. Aporte y limitaciones</b> .....	18
<b>3.3.1.7. Definiciones, Siglas y Abreviaciones</b> .....	19
<b>CAPITULO IV ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE</b> .....	21
<b>4. Especificación de Requerimientos de Software</b> .....	22
<b>4.1. Alcances</b> .....	22
<b>4.2. Objetivos del software</b> .....	22
<b>4.2.1. Objetivo General</b> .....	22
<b>4.2.2. Objetivos Específicos</b> .....	23
<b>4.3. Descripción global del producto</b> .....	23
<b>4.3.1. Interfaces con el usuario</b> .....	23
<b>4.3.2. Interfaces con el hardware</b> .....	26
<b>4.3.3. Interfaces con el software</b> .....	43

4.3.4.	<b>Interfaces de comunicaciones</b> .....	44
4.4.	<b>Requerimientos específicos</b> .....	44
4.4.1.	<b>Requerimientos funcionales del sistema web</b> .....	44
4.4.2.	<b>Interfaces externas de entrada</b> .....	45
4.4.3.	<b>Interfaces externas de salida</b> .....	46
4.5.	<b>Factibilidad del proyecto</b> .....	46
4.5.1.	<b>Factibilidad técnica</b> .....	46
4.5.2.	<b>Factibilidad operacional</b> .....	48
4.5.3.	<b>Factibilidad económica</b> .....	49
<b>CAPITULO V IMPLEMENTACIÓN</b> .....		52
5.	<b>Implementación</b> .....	53
5.1.	<b>Diseño</b> .....	53
5.1.1.	<b>Modelo entidad Relación</b> .....	53
<b>CAPITULO VI PRUEBAS Y RESULTADOS</b> .....		59
6.	<b>Pruebas y Resultados</b> .....	60
<b>CAPITULO VII CONCLUSIONES</b> .....		63
<b>Bibliografía</b> .....		65
<b>Anexos</b> .....		66

## Índice Tablas

<b>Tabla n° 1:</b> Características Modulo Node MCU Lolin v3.....	26
<b>Tabla n° 2:</b> Pines de conexión lado izquierdo Node MCU Lolin v3.....	27
<b>Tabla n° 3:</b> Pines de conexión lado derecho Node MCU Lolin v3. ....	28
<b>Tabla n° 4:</b> Especificaciones técnicas Arduino pro micro.....	30
<b>Tabla n° 5:</b> Pines de alimentación Arduino pro micro. ....	31
<b>Tabla n° 6:</b> Características sensor ultrasonidos HC-SR04 .....	33
<b>Tabla n° 7:</b> Pines de alimentación sensor ultrasonidos HC-SR04.....	34
<b>Tabla n° 8:</b> Lado de mayor tensión Convertidor de niveles 3.3 / 5 V .....	35
<b>Tabla n° 9:</b> Lado de menor tensión Convertidor de niveles 3.3 / 5 V .....	36
<b>Tabla n° 10:</b> Requerimientos funcionales del sistema web .....	44
<b>Tabla n° 11:</b> Interfaces externas de entrada.....	45
<b>Tabla n° 12:</b> Interfaces externas de salida .....	46
<b>Tabla n° 13:</b> Requerimientos técnicos del sistema web .....	47
<b>Tabla n° 14:</b> Requerimientos técnicos de la programación del IDE-Arduino.....	47
<b>Tabla n° 15:</b> Evaluación económica .....	50
<b>Tabla n° 16:</b> Costos recursos humanos.....	51
<b>Tabla n° 17:</b> Detalles de prueba realizada correspondiente al mes de enero 2019.....	60
<b>Tabla n° 18:</b> Resultados obtenidos correspondiente al mes de enero 2019.....	61

## Índice Figuras

<b>Figura n° 1:</b> Imagen del Sistema web visualizando menú principal. ....	24
<b>Figura n° 2:</b> Imagen del Sistema web visualizando la hora que I / S vehículos en un día..	24
<b>Figura n° 3:</b> Imagen del Sistema web visualizando el día que I / S vehículos en un mes...	25
<b>Figura n° 4:</b> Imagen visualizando la cantidad de vehículos que I / S en un año .....	25
<b>Figura n° 5:</b> Modulo WIFI Node MCU Lolin v3. ....	26
<b>Figura n° 6:</b> Chip integrado ESP8266 al Node MCU Lolin v3.....	29
<b>Figura n° 7:</b> Imagen referencial de un Arduino pro micro.....	29
<b>Figura n° 8:</b> Imagen referencial conexiones Arduino pro micro.....	31
<b>Figura n°9:</b> Imagen referencial de una Protoboard. ....	32
<b>Figura n° 10:</b> Sensor HC-SR04 .....	33
<b>Figura n° 11:</b> Convertidor niveles 3.3 V / 5 V y viceversa. ....	35
<b>Figura n° 12:</b> Cables eléctricos. ....	37
<b>Figura n° 13:</b> Circuito Electrónico (PCB). ....	38
<b>Figura n° 14:</b> Vista general de integracion de microcontroladores.....	39
<b>Figura n° 15:</b> Conexion Arduino micro a sensores HC-SR04 .....	40
<b>Figura n° 16:</b> Calculo distancia Sensor HC-SR04 .....	41
<b>Figura n° 17:</b> Conexión del Node MCU Lolin v3 a sistema web. ....	41
<b>Figura n° 18:</b> Node MCU Lolin v3 – Conversor 3.3 V / 5 V y Arduino pro micro.....	42
<b>Figura n° 19:</b> Menú principal del sistema web.....	43
<b>Figura n° 20:</b> Sketch de visualización Software IDE-Arduino .....	48
<b>Figura n° 21:</b> Modelo Entidad Relación ocupado para el sistema .....	53
<b>Figura n° 22:</b> Base de datos a ocupar en el sistema. ....	54
<b>Figura n° 23:</b> Interfaz para acceso a gráficos. ....	55
<b>Figura n° 24:</b> Grafico 1, corresponde a vehículos / día.....	56
<b>Figura n° 25:</b> Grafico 2, corresponde a vehículos / mes. ....	57
<b>Figura n° 26:</b> Grafico 3, corresponde a vehículos / año.....	58
<b>Figura n° 27:</b> Grafico del sistema web correspondiente al mes de enero 2019 .....	61



# **CAPÍTULO I INTRODUCCIÓN**

## **1.1.Introducción**

En la actualidad, se puede reflejar el crecimiento del flujo vehicular en la Universidad del Bío-Bío con la poca disponibilidad de estacionamientos dentro del campus Concepción, no existiendo control y monitoreo de autos que ingresan y salen de esta institución.

Es por aquello, que el presente proyecto tiene por objetivo crear un Sistema Cyber-Físico, que pueda implementar la visualización en tiempo real de vehículos que ingresan y salen dentro de la Universidad del Bío-Bío, mediante la utilización de sensores y microcontroladores que envíen datos a través de la placa ESP8266. Esta información es visualizada por medio de un sistema web, detallando la cantidad total del flujo de vehículos, indicando la fecha correspondiente y horarios que ingresaron y salieron del campus.

Este proyecto se diferencia de los actuales controles de acceso de vehículos que se encuentran en centros comerciales y grandes tiendas por su bajo costo económico y la tecnología utilizada, ya que el sistema desarrollado no implica una inversión monetaria importante, siendo los arduinos de libre acceso, pudiendo ser conectados a internet y ser optimizados a la hora de ser programados. Es un factor clave a la hora de realizar el desarrollo de este proyecto ya que en una etapa posterior tanto administrativos como futuros estudiantes de la Universidad del Bío-Bío puedan hacer uso del sistema, con el fin de saber en tiempo real si existe disponibilidad de estacionamientos sin necesidad de hacer ingreso de forma presencial al campus Concepción.

Dentro de las ventajas que se van a obtener al desarrollar este sistema es que la Universidad del Bío-Bío pueda contar con un nuevo sistema de control y monitoreo, y así generar informes a base de estadísticas de los resultados obtenidos, a diferencia del actual control que se tiene con barreras de acceso y personal de seguridad.

El proyecto deja una base para una etapa a futuro poder integrar el actual sistema de barreras que controla los accesos a la Universidad del Bío-Bío con el sistema Cyber-Físico, así automatizando su funcionamiento y evitando el gran flujo vehicular, teniendo un control en el ámbito de seguridad para alumnos y administrativos que hacen uso de estacionamientos.

# **CAPITULO II DEFINICIÓN DE LA INSTITUCIÓN**

## 2. Definición de la institución

### 2.1. Descripción de la institución

- **Antecedentes generales:**
  - **Nombre:** Universidad del Bío-Bío, campus Concepción.
  - **Dirección:** Avenida Collao #1202, Concepción.
  - **Rubro:** Institución de educación superior
  - **Teléfono:** Casilla 5-C - CP: 4051381. +56-413111200.

- **Misión**

Ser reconocida a nivel nacional e internacional como una Universidad pública, responsable socialmente y regional que, comprometida con su rol estatal, desde las Regiones del Biobío y Ñuble, forma personas integrales de excelencia y aporta a través de su quehacer al desarrollo sustentable de las regiones y el país. ([http://www.ubiobio.cl/w/index\\_normal.php#Vision\\_y\\_Mision](http://www.ubiobio.cl/w/index_normal.php#Vision_y_Mision), 04 de enero del 2019).

- **Visión**

La Universidad del Bío-Bío, a partir de su naturaleza pública, responsable socialmente y estatal, tiene por misión, desde las Regiones del Biobío y Ñuble, aportar a la sociedad con la formación de personas integrales, a través de una Educación Superior de excelencia. Comprometida con los desafíos regionales y nacionales, contribuye a la movilidad e integración social por medio de la generación y transferencia de conocimiento avanzado, mediante la docencia de pregrado y postgrado de calidad, la investigación fundamental, aplicada y de desarrollo, la vinculación bidireccional con el medio, la formación continua y la extensión. Asimismo, impulsa el emprendimiento y la innovación, el fortalecimiento de la internacionalización y el desarrollo sustentable de sus actividades, basada en una cultura participativa centrada en el respeto a las personas ([http://www.ubiobio.cl/w/index\\_normal.php#Vision\\_y\\_Mision](http://www.ubiobio.cl/w/index_normal.php#Vision_y_Mision), 04 de enero del 2019).

## **2.2.Descripción del área de estudio**

El proyecto es desarrollado para la Universidad del Bío-Bío en su campus Concepción, y principalmente el estudio incluye 2 áreas de desarrollo. La primera contempla la integración de sensores, microcontroladores y manejo de arduino con modulo wifi ESP8266. En tanto la segunda área abarca el desarrollo web bajo React JS, este sistema cumple la función de visualización de gráficos del flujo vehicular, mediante la elaboración de una interfaz de programación se captura los datos que envían los sensores y se despliegan de forma visual en el sistema.

## **2.3.Descripción de la problemática**

En la actualidad, no existe ningún tipo de control y registro del flujo vehicular dentro de los distintos accesos a estacionamientos emplazados en el campus Concepción de la Universidad del Bío-Bío, lo que conlleva al bajo control y el no disponer de exactitud de vehículos que ingresan en una determinada fecha.

Esto genera que no se puedan hacer planes de contingencia ni obtener estadísticas de su uso, no pudiendo optimizar su capacidad de ninguna forma, lo cual hoy en día se ve reflejado en el gran tráfico vehicular que se ve afectada la Universidad del Bío-Bío, en donde no existe un orden y control al momento del ingreso y salida vehicular.

# **CAPITULO III DEFINICIÓN DEL PROYECTO**

### **3. Definición del proyecto**

#### **3.1.Objetivo general del proyecto**

Construir un sistema Cyber-Físico que permita el monitoreo en tiempo real, de entrada y salida vehicular en la Universidad del Bío-Bío, con el objetivo de obtener estadísticas e información relevante asociado a la congestión vehicular que afronta actualmente la Universidad en su campus Concepción.

#### **3.2.Objetivos específicos del proyecto**

- Entregar una herramienta que permita conocer y obtener estadísticas en tiempo real de los vehículos que hacen ingreso a las instalaciones de la Universidad del Bío-Bío.
- Facilitar el acceso a la data generada que esta se presente de manera intuitiva y fácil de leer para quien la necesite.
- Analizar el actual control vehicular en la Universidad del Bío-Bío de tal manera de verificar falencias de dicho servicio.
- Construir el hardware necesario para la detección del ingreso y salida de los vehículos en el campus Concepción.
- Desarrollar un sistema que permita la comunicación en tiempo real del hardware ubicado en los estacionamientos, del mismo modo con el servidor donde se almacenan los datos.
- Proveer información relevante a base de estadísticas del monitoreo vehicular para su posterior utilización en la toma de decisiones.

### **3.3.Lenguaje, herramientas y recursos**

A continuación, se presentan el ambiente de ingeniería de software ocupado para el desarrollo del sistema Cyber-Físico.

#### **3.3.1. Ambiente de Ingeniería de Software**

##### **3.3.1.1.Metodología**

El desarrollo de este proyecto Cyber-Físico, contempla la elaboración y entrega iterativa e incremental del sistema. se consideró un total de tres incrementos y se escoge esta metodología en función del proyecto, ya que esta metodología permite realizar una evolución sostenida del objetivo a alcanzar, prácticamente desde el primer momento.

- **Incremento 1:** Esta primera etapa contempla la configuración e integración de sensores de distancia, programación de microcontroladores, construcción de contenedores de sensores y el diseño del circuito impreso (PCB) a ocupar en el sistema.
- **Incremento 2:** En esta etapa se tiene en consideración la topología de red a utilizar, que será una comunicación inalámbrica por medio de wifi del microcontrolador Node NCU Lolin v3 (ESP8266). Los protocolos de comunicación serán persistentes en tiempo real para envíos de datos de forma inalámbrica.
- **Incremento 3:** Establecer el desarrollo final del proyecto, por medio de un sistema web, para el almacenamiento de datos que arrojen los sensores al detectar un vehículo, esta información se almacena en el servidor virtual gestionado con Telefónica del Sur para el desarrollo del proyecto.



### 3.3.1.2. Tecnologías a utilizar

A continuación, los conceptos que contempla el desarrollo de este sistema:

- **Sistema Cyber-Físico**  
Es un mecanismo (sistema físico) controlado o monitoreado por algoritmos basados en computación y estrechamente integrados con internet. (Wikipedia)
- **IDE-Arduino**  
De código abierto (IDE) facilita escribir código y cargarlo. El entorno está escrito en Java y se basa en procesamiento. (Arduino.cc)
- **React**  
Es una biblioteca javascript de código abierto diseñada para crear interfaces de usuario con el objetivo de facilitar el desarrollo de aplicaciones en una sola página. (Wikipedia)
- **Persistencia**  
En programación, la persistencia es la acción de preservar la información de un objeto de forma permanente (guardado), pero a su vez también se refiere a poder recuperar la información del mismo (leerlo) para que pueda ser nuevamente utilizado. (Wikipedia)
- **Base de datos relacional:**  
La base de datos relacional (BDR) es un tipo de base de datos (BD) que cumple con el modelo relacional, el modelo más utilizado actualmente para implementar el BD ya planificadas. (Wikipedia)

### 3.3.1.3. Técnicas y notaciones

Para la toma de requerimientos se realizaron reuniones de colaboración con el docente sr. Carlos Araneda, quien dicta actualmente electivo de Arduinos para las carreras de Ingeniería Civil en Informática e Ingeniería de Ejecución en Computación e Informática. Estas reuniones abarcaban principalmente el describir la problemática actual, para ello fueron esenciales los conocimientos que entrego Carlos Araneda, de acuerdo a esto se buscó las posibles soluciones al momento de desarrollar el sistema y la mejor manera de poder afrontar los problemas que esto conlleva.

#### **3.3.1.4. Estándares de documentación**

- Solicitud de inscripción de actividad de titulación proyecto de título.
- Normativa del sistema de bibliotecas WERKEN de la Universidad del Bío-Bío.
- Plantilla de redacción para hacer uso de referencias bibliográficas.
- IEEE software test documentation Std 829-1998.
- Plantilla de tesis, Universidad del Bío-Bío.

#### **3.3.1.5. Herramientas de apoyo**

En esta sección se describen los diferentes software que fueron necesarios para el desarrollo del sistema Cyber-Físico.

Para el desarrollo del informe fue necesario el manejo de:

- Microsoft Office 365

Para el desarrollo y programación del sistema web:

- Sublime Text 3.
- Notepad++
- Visual Studio

Para la programación e integración de microcontroladores, sensores y arduino fue necesario el manejo del software:

- IDE-Arduino

#### **3.3.1.6. Aporte y limitaciones**

Dentro de los aportes que presenta este sistema se cuenta con:

- Que la Universidad del Bío-Bío pueda contar con un nuevo sistema de control y monitoreo de entrada y salida de vehículos dentro del campus Concepción.
- Que la universidad genere informes en base a resultados obtenidos del sistema web en un rango de tiempo.

- Obtener estadísticas en base al historial que se pueda encontrar, ya sea por día, mes y año determinado.
- Que la actual seguridad que rige en la Universidad del Bío-Bío puedan identificar los horarios de mayor flujo de vehículos y así realizar un plan de contingencia.

Por otra parte, este proyecto tiene algunas limitaciones las cuales son:

- Este proyecto no contempla la integración con el actual acceso de barreras que cuenta la Universidad del Bío-Bío.
- Falta de cobertura WIFI en la conexión a internet desde los accesos a la Universidad del Bío-Bío.
- No hay exactitud de la cantidad total de estacionamientos disponibles en el campus Concepción.
- Falta de personal fijo que pueda hacer uso y monitorear el sistema, evitando posible duplicación de información o un registro erróneo.
- Ubicación de sensores evitando el tráfico de personas para lograr con exactitud el control de vehículos.

### 3.3.1.7. Definiciones, Siglas y Abreviaciones

Definiciones generales orientadas al sistema web:

- **PHP (Hypertext Pre-Processor)** Lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML. (PHP.net)
- **HTML (HyperText Markup Language)** Es un lenguaje de marcado para el desarrollo de páginas web. Define una estructura básica y un código para la definición de contenido de una página web, como texto, imágenes, videos, entre otros (Wikipedia, e).
- **SQL (Structured Query Language)** Es un lenguaje de programación diseñado para almacenar, manipular y recuperar datos almacenados en bases de datos relacionales.

- **MySQL** Es un sistema de gestión de base de datos relacional desarrollado bajo licencia dual: Publica general / Licencia comercial por Oracle (Wikipedia).
- **JavaScript** Lenguaje de programación que abarca principalmente efectos atractivos y dinámicos en sitios web.
- **I / S** Ingreso y Salida de vehículos.

Definiciones generales orientadas al sistema físico:

- **IDE-Arduino** Software de programación bajo el cual fue desarrollado el sistema físico.
- **Arduino** Es una compañía de software y hardware de código abierto.
- **Microcontrolador** Circuito integrado programable.
- **Sensores ultrasonido** Dispositivo que mide la distancia mediante el uso de ondas ultrasónicas.
- **ESP8266** Chip que proporciona WIFI y viene ensamblado al Node MCU Lolin v3.
- **ATmega32U4** Microcontrolador de 8 bits de bajo consumo, ejecutando instrucciones en un solo ciclo de reloj.
- **GPIO** Entrada y salida de propósito general.
- **GND** Conexión a tierra.
- **VCC** Voltaje que puede variar según el componente entre 3.3 V y 5 V.
- **RAW** Pin es la entrada de voltaje no regulada.
- **RST** Sirve para reiniciar el chip por completo.
- **V** Tensión eléctrica (Voltaje).
- **ECHO** Devuelve un pulso al microcontrolador, de tal manera de saber el tiempo que demora en retornar desde el objeto a medir y el sensor.
- **TRIG** Recibe un pulso, el cual indica al módulo que comience a calcular la distancia.
- **PCB** Placa de circuito impreso.
- **GNU** Licencia Publica General.

# **CAPITULO IV ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE**

## **4. Especificación de Requerimientos de Software**

A continuación, se presentará la propuesta de construcción del sistema Cyber-Físico para monitoreo de entrada y salida de vehículos en la Universidad del Bío-Bío, bajo los aspectos y definiciones que dicta la Ingeniería de Software.

### **4.1. Alcances**

Este proyecto contempla la creación de un sistema Cyber-Físico, permitiendo al usuario la visualización de gráficos y registros de la cantidad de vehículos que ingresan y salen de la Universidad del Bío-Bío en una determinada fecha ingresada.

Lo que se propone en este proyecto a desarrollar, busca solucionar esta falta de información, al momento de tener un control de las entradas en el campus, y así utilizar de forma tal que permita un mejor uso de los estacionamientos disponibles, gracias a que la información está en tiempo real en el sistema, enviando reportes constantemente del flujo vehicular, siendo relevante los rangos de tiempos en donde se producen el mayor flujo del parque vehicular.

Este proyecto si bien no contempla el control de acceso ocupando barreras deja la base necesaria para que estas se puedan integrar en una etapa posterior, permitiendo implementación de un estacionamiento inteligente, completamente automatizado dentro de la Universidad del Bío-Bío. Este proyecto puede desembocar en una herramienta mecanizada que permita a los usuarios de los estacionamientos, saber con anterioridad si es que hay espacio o no, sin la necesidad de llegar y no encontrar estacionamientos disponibles.

### **4.2. Objetivos del software**

#### **4.2.1. Objetivo General**

Desarrollar y construir un sistema físico y web de monitoreo de vehículos en los accesos a la Universidad del Bío-Bío, obteniendo información verídica del flujo vehicular dentro del campus Concepción.

#### **4.2.2. Objetivos Específicos**

- Detectar cantidad de autos que ingresaron / salieron de la Universidad del Bío-Bío en el rango de hora / mes y año.
- Visualizar de manera gráfica en tiempo real el monitoreo y control vehicular en la Universidad del Bío-Bío.

#### **4.3.Descripción global del producto**

##### **4.3.1. Interfaces con el usuario**

El marco en el cual interactúa el usuario con el sistema Cyber-Físico, es mediante una plataforma web para visualizar gráficos que está alojado en servidor virtual de Telefónica del Sur, dando la posibilidad a los usuarios el ingresar sin importar el lugar donde se encuentren. Dentro de los gráficos que podemos encontrar, se diferencian en el detalle a mostrar, ya sea por día, mes y año. El despliegue de barras graficas se diferencia en dos colores, uno azul que corresponde a vehículos de entrada y otro rojo de salida que hace referencia a los vehículos que salen del campus Concepción.

Las siguientes figuras muestra el sistema web que visualiza el usuario al momento de ingresar:

**Figura n° 1:** Imagen del Sistema web visualizando menú principal.



**Fuente:** Elaboración propia – Sistema web

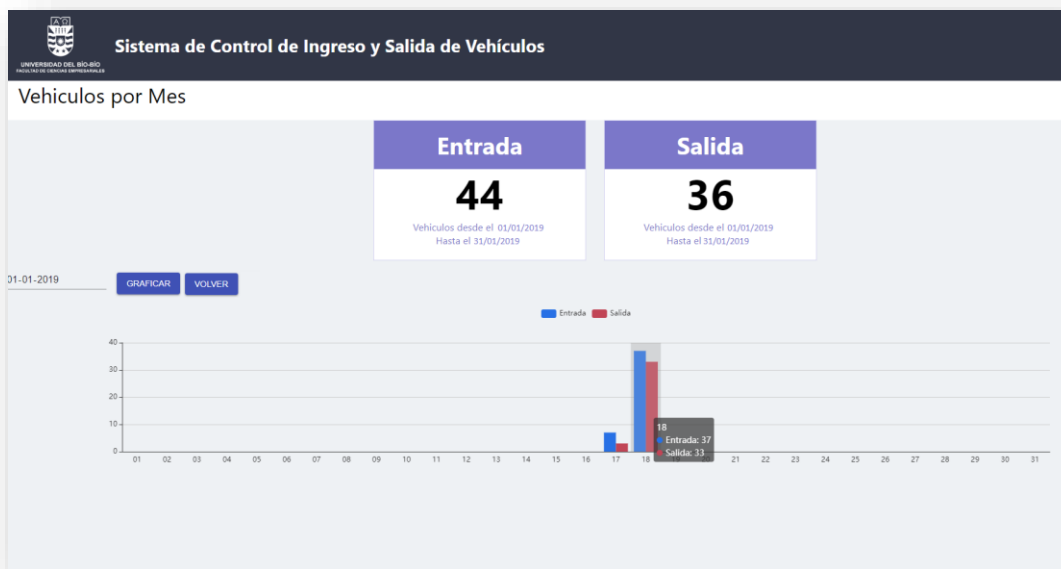
**Figura n° 2:** Imagen del Sistema web visualizando la hora que I / S vehículos en un día.



**Fuente:** Elaboración propia – Sistema web



**Figura n° 3:** Imagen del Sistema web visualizando el día que I / S vehículos en un mes.



**Fuente:** Elaboración propia – Sistema web

**Figura n° 4:** Imagen visualizando la cantidad de vehículos que I / S en un año.



**Fuente:** Elaboración propia – Sistema web

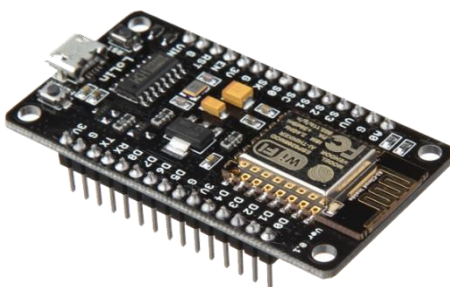
### 4.3.2. Interfaces con el hardware

Descripción de dispositivos para desarrollar Sistema Físico:

- **Modulo WIFI Node MCU Lolin v3 (ESP8266)**

Es una tarjeta de desarrollo similar a un arduino, ya que incorpora modulo WIFI para su conexión inalámbrica. La principal función del módulo será otorgar WIFI a nuestro sistema físico, y así enviar la data a la base de datos que estará alojada en el servidor virtual dispuesto para el desarrollo.

*Figura n° 5: Modulo WIFI Node MCU Lolin v3.*



*Fuente: <https://tienda.bricogeek.com/wifi/1033-nodemcu-v3-esp8266.html>*

**La tabla 1 muestra las características generales Modulo Node MCU Lolin v3:**

**Tabla n° 1:** Características Modulo Node MCU Lolin v3

<b>Procesador</b>	ESP8266 80Mhz 3.3V
<b>Memoria</b>	4MB Flash
<b>WIFI</b>	802.11 b/g/n
<b>Regulador</b>	3.3V integrado 500mA
<b>Voltaje USB</b>	5V

<b>Voltaje de Entradas/Salidas</b>	3.3V
<b>Pines</b>	17 pines de entrada y salida GPIO
<b>Entrada externa alimentación</b>	20V máximo

*Fuente: Elaboración propia*

Distribución lado izquierdo del módulo:

**Tabla n° 2:** Pines de conexión lado izquierdo Node MCU Lolin v3.

<b>A0</b>	Canal de propósito general
<b>RSV</b>	Reservado
<b>RSV</b>	Reservado
<b>SD3</b>	Canal de propósito general
<b>SD2</b>	Canal de propósito general
<b>SD1</b>	Canal de propósito general
<b>CMD</b>	Canal de propósito general
<b>SDO</b>	Canal de propósito general
<b>CLK</b>	Canal de propósito general
<b>GND</b>	Conexión a tierra
<b>3V3</b>	Voltaje
<b>EN</b>	Sistema
<b>RST</b>	Sistema
<b>GND</b>	Conexión a tierra
<b>VIN</b>	5V Encendido

*Fuente: Elaboración propia*

Distribución lado derecho del módulo:

**Tabla n° 3:** Pines de conexión lado derecho Node MCU Lolin v3.

<b>D0</b>	Canal de propósito general
<b>D1</b>	Canal de propósito general
<b>D2</b>	Canal de propósito general
<b>D3</b>	Canal de propósito general
<b>D4</b>	Canal de propósito general
<b>3V3</b>	Voltaje
<b>GND</b>	Conexión a tierra
<b>D5</b>	Canal de propósito general
<b>D6</b>	Canal de propósito general
<b>D7</b>	Canal de propósito general
<b>D8</b>	Canal de propósito general
<b>D9</b>	Canal de propósito general
<b>D10</b>	Canal de propósito general
<b>GND</b>	Conexión a tierra
<b>3V3</b>	Voltaje

*Fuente: Elaboración propia*

**Figura n° 6:** Chip integrado ESP8266 al Node MCU Lolin v3.

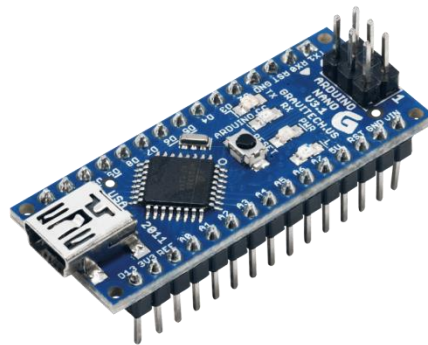


**Fuente:** <https://programarfacil.com/podcast/nodemcu-tutorial-paso-a-paso/>

- **Arduino pro micro**

Es una pequeña placa programable basada en el ATmega32U4 con conexión micro USB, lo cual beneficia para trabajar directo desde un computador.

**Figura n° 7:** Imagen referencial de un Arduino pro micro.



**Fuente:** <http://madrigalelectronics.com/arduino-pro-mini-5v-generico.html>

**Especificaciones técnicas generales del Arduino pro micro:**

**Tabla n° 4:** Especificaciones técnicas Arduino pro micro

<b>Microcontrolador</b>	ATmega32U4
<b>Tensión de funcionamiento</b>	5 V
<b>Voltaje de entrada (recomendado)</b>	7-12V
<b>Voltaje de entrada (límite)</b>	6-20 V
<b>Pines digitales I/O</b>	20
<b>Canales PWM</b>	7
<b>Canales de entrada analógicos</b>	12
<b>Corriente continua para Pin I / O</b>	20 mA
<b>Corriente CC para Pin 3.3V</b>	50 mA
<b>Memoria flash</b>	32KB
<b>SRAM</b>	2,5 KB
<b>EEPROM</b>	1 KB
<b>Velocidad de reloj</b>	16 MHz
<b>Longitud</b>	48 mm
<b>Anchura</b>	18 mm
<b>Peso</b>	13 g

*Fuente: Elaboración propia.*

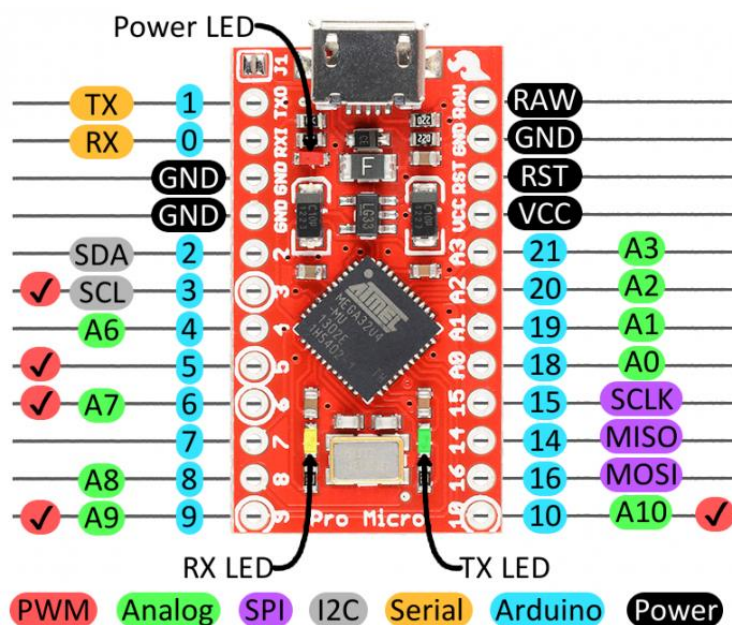
**Tabla n° 5:** Pines de alimentación Arduino pro micro.

VLN	Voltaje de entrada
5 V	Fuente de alimentación para alimentar microcontrolador y otros componentes.
3 V	Suministro de 3,3, funciona de la misma manera que los 5 V pero regulador a menor tamaño.
GND	Pines de tierra.

*Fuente: Elaboración propia.*

**Esquema general de entradas y salidas Arduino pro micro:**

*Figura n° 8: Imagen referencial conexiones Arduino pro micro*

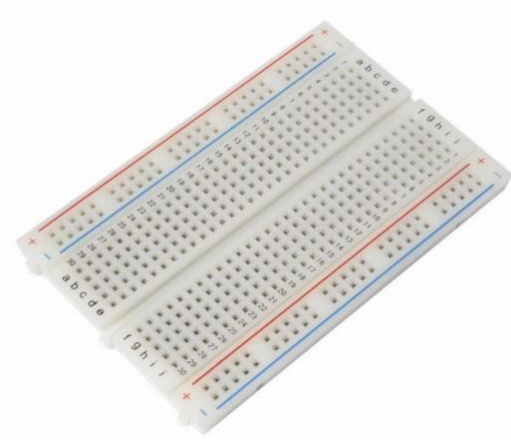


*Fuente: <https://saber.patagoniatec.com/2014/09/arduino-pro-micro-atmega32u4-arduino-argentina-ptec-leonardo/>*

- **RAW** Pin es la entrada de voltaje no regulada.
  - **VCC** Esta es la entrada que alimenta al arduino y sus componentes.
  - **RST** Sirve para reiniciar el chip por completo.
  - **GND** Tierra.
  - Los pines A0 – A3 pueden ser usados como Entradas / Salidas digitales.
  - Los pines digitales 4 – 6 – 8 – 9 – 10 pueden ser usados como analógicos.
- **Protoboard**

Es ampliamente usada en diseño y experimentos de proyectos de sistemas digitales, dentro de la variedad de las protoboard se pueden encontrar de distintos tamaños de distribución, así mismo con varios puntos de alimentaciones divididas en distintas filas y columnas.

*Figura n°9: Imagen referencial de una Protoboard.*



*Fuente: <http://i2celectronica.com/protoboards-placas-pcb/98-protoboard-400-puntos.html>*

- **Líneas azules:** Corresponden a pines de alimentaciones positivo (VCC).
- **Líneas rojas:** Corresponden a pines de tierra (GND).
- **Parte central:** Corresponden a los diversos puntos de conexión de entrada / salidas a dispositivos.



- **Sensor ultrasonidos HC-SR04**

Los sensores de distancia como se refleja en la figura n° 10, permiten principalmente detectar movimiento, que hace que varié la frecuencia de la onda al rebotar en el objeto en movimiento el cual está dentro del rango del sensor. El rango de estos sensores se encuentra dentro de los 4.5 metros como máximo y 1.7 cm como mínimo.

*Figura n° 10: Sensor HC-SR04*



*Fuente: <https://electronilab.co/tienda/sensor-de-distancia-de-ultrasonido-hc-sr04/>*

**Características generales del sensor ultrasonido HC-SR04:**

**Tabla n° 6: Características sensor ultrasonidos HC-SR04**

<b>Dimensiones del sensor</b>	43 x 20 x 17 mm
<b>Tensión de alimentación</b>	5 V
<b>Frecuencia de trabajo</b>	40 KHz
<b>Rango máximo</b>	4.5 m
<b>Rango mínimo</b>	1.7 cm
<b>Duración mínima de disparo</b>	10 uS
<b>Duración de pulso salida</b>	100-25000 uS

<b>Tiempo mínimo de espera entre medidas</b>	20 mS
--	-------

*Fuente: Elaboración propia.*

**Tabla n° 7:** Pines de alimentación sensor ultrasonidos HC-SR04

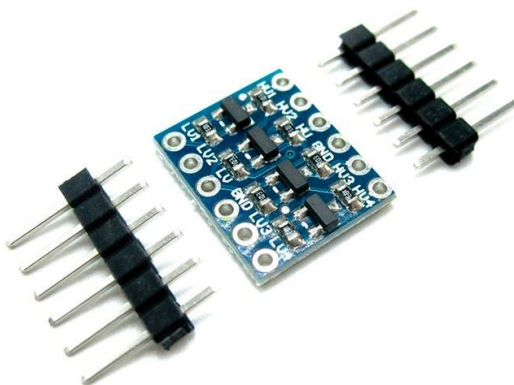
<b>VCC</b>	Voltaje
<b>TRIG</b>	Disparo del ultrasonido
<b>ECHO</b>	Recepción del ultrasonido
<b>GND</b>	Conexión a tierra

*Fuente: Elaboración propia.*

- **Convertidor de niveles 3.3 / 5 V**

Es un pequeño dispositivo electrónico que baja los voltajes lógicos, transforma de 5V a 3.3V y de 3.3V a 5V, es un dispositivo muy útil en un procesador como arduino. Este dispositivo es de gran utilidad al momento de implementar el sistema físico, ya que la mayoría de procesadores operan a distintos voltajes, siendo los más habituales 5V y 3.3V. Hoy en día los arduinos modernos operan a 5V, en cambio el resto de los dispositivos electrónicos (sensores, controladores, modulo WIFI, etc.) operan a distintas tensiones, ya sean 5V o 3.3V, es por aquello que es necesario ocupar el conversor de acuerdo a los microcontroladores que se ocupan en este sistema.

**Figura n° 11:** Convertidor niveles 3.3 V / 5 V y viceversa.



**Fuente:** <https://afel.cl/producto/conversor-de-nivel-logico-i2c-5v-a-3-3v/?v=5bc574a47246>

**Características generales del Convertidor niveles 3.3 V / 5 V:**

La conexión del convertidor de niveles es bastante simple, cuenta con dos lados distintos, uno para el dispositivo de mayor tensión (nombrado por HIGH en el convertidor) y uno para el dispositivo de menor tensión (nombrado por LOW en el convertidor).

Pines de conexión del lado de mayor tensión:

**Tabla n° 8:** Lado de mayor tensión Convertidor de niveles 3.3 / 5 V

<b>HV4</b>	GPIO	Canal 4
<b>HV3</b>	GPIO	Canal 3
<b>GND</b>	GND	Conexión a Tierra
<b>HV</b>	HV	VCC mayor tensión
<b>HV2</b>	HV2	Canal 2
<b>HV1</b>	HV1	Canal 1

**Fuente:** Elaboración propia.

- LV entre 1.5V y 7V.
- Canales funcionan tanto para señales digitales como análogas.
- GND debe ser común para ambos dispositivos.

Pines de conexión del lado de menor tensión:

**Tabla n° 9:** Lado de menor tensión Convertidor de niveles 3.3 / 5 V

<b>LV4</b>	<b>GPIO</b>	<b>Canal 4</b>
<b>LV3</b>	<b>GPIO</b>	<b>Canal 3</b>
<b>GND</b>	<b>GND</b>	<b>Conexión a Tierra</b>
<b>LV</b>	<b>3.3V</b>	<b>VCC menor tensión</b>
<b>LV2</b>	<b>GPIO</b>	<b>Canal 2</b>
<b>LV1</b>	<b>GPIO</b>	<b>Canal 1</b>

*Fuente: Elaboración propia.*

- HV entre LV y 15V
- Canales funcionan tanto para señales digitales como análogas.
- GND debe ser común para ambos dispositivos.

- **Accesorios cables**

Elemento fabricado para conducir electricidad, son principalmente realizados en cobre.

Pueden ser combinaciones:

- Macho / macho
- Macho / hembra
- Hembra / hembra

*Figura n° 12: Cables eléctricos.*



**Fuente:** <https://electronilab.co/tienda/65-cables-conectores-para-protoboard-macho-macho/>

- **PCB (Printed Circuit Board)**

Es una superficie constituida por caminos, pistas o buses de material conductor laminadas sobre una base no conductora (Wikipedia, 2015).

Este Circuito Electrónico se implementó principalmente para evitar el uso de protoboard y la variedad de cables al momento de ensamblar todo el sistema físico.

**Figura n° 13:** Circuito Electrónico real diseñado para el Sistema Físico

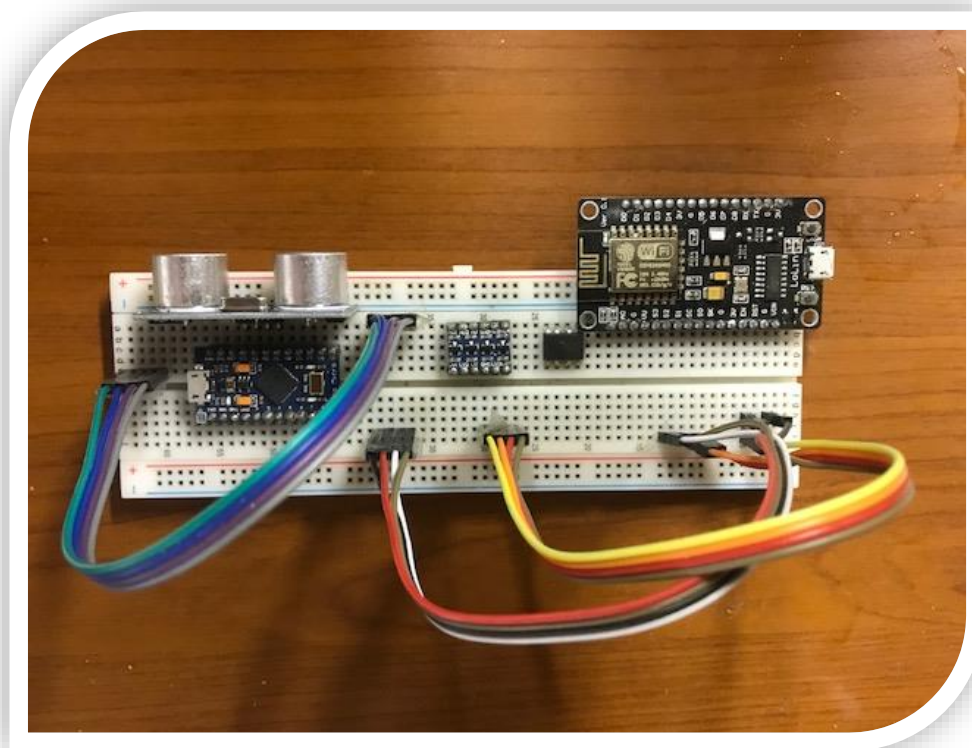


**Fuente:** Diseño propio – Fabricación MCI Electronics

**Principalmente la conexión que tendrá el lado físico al sistema web del proyecto a desarrollar será:**

- Estructura general de integración de sensor de distancia, node MCU Lolin v3, convertidor 3.3 V a 5 V y arduino pro micro en protoboard. Este primer diseño quedo denominado como la versión 1.2 en el desarrollo del sistema.

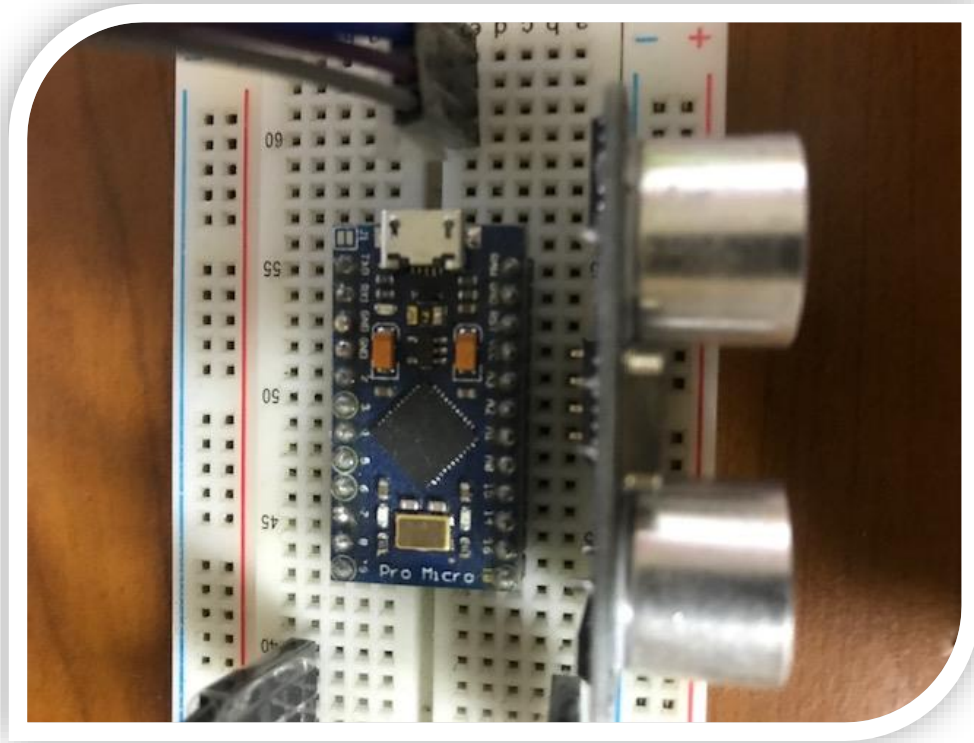
*Figura n° 14: Vista general de integracion de microcontroladores.*



*Fuente: Elaboración propia – Sistema físico.*

- Vinculación de los dos sensores ultrasonido HC-SR04 con las respectivas entradas del arduino pro micro en protoboard.

**Figura n° 15:** *Conexion Arduino pro micro a sensor HC-SR04*



**Fuente:** *Elaboración propia – Sistema físico.*

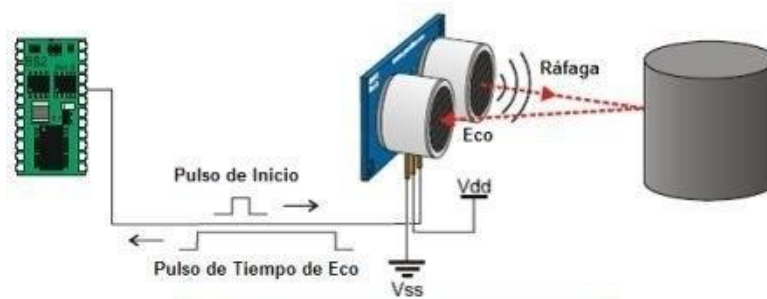
- Vinculación de los dos sensores ultrasonido HC-SR04 al Arduino pro micro.
- Salidas sensor ultrasonido HC-SR04: VCC – TRIG – ECHO – GND



### Ejemplo de captura de distancia usando sensor HC-SR04:

El sensor emite una señal dentro del rango de 2 a 450cm, el objeto a calcular la distancia serán los vehículos que se ubiquen dentro del rango establecido. El pulso de retorno llega al Arduino pro micro otorgando los datos necesarios para que el Node MCU transmita los respectivos datos.

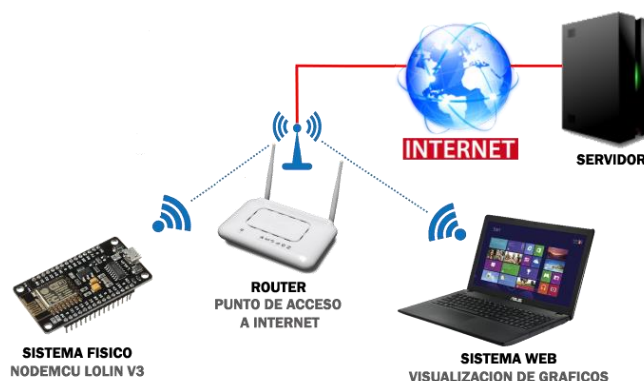
**Figura n° 16:** Calculo distancia Sensor HC-SR04



**Fuente:** <https://amgkits.com/home/9-sensor-ultrasonico-hc-sr04.html>

- Por medio del conversor se conecta al arduino otorgando un sistema homogéneo al momento de ejecutar el código. El modulo WIFI envía la información a la base de datos que está alojada en el servidor virtual con IP (190.121.25.248).

**Figura n° 17:** Conexión del Node MCU Lolin v3 a sistema web.



**Fuente:** <http://www.esploradores.com>

- Estructura final del sistema quedo denominada como versión 1.3 y ya cuenta con la integración de la PCB y los microcomponentes asociados al sistema físico.

**Figura n° 18:** Vista general del sistema Físico en PCB.



**Fuente:** Elaboración propia – Sistema físico.

- El sistema web principalmente otorga la visualización de los gráficos que recopila la información que se encuentran en la base de datos, con gestor MySQL y estará alojada en servidor virtual de Telefónica del Sur. En las siguientes imágenes se ve el sistema web con menú principal abarca los tres gráficos señalados, ya sea por día, mes y año.

*Figura n° 19: Menú principal del sistema web*



*Fuente: Elaboración propia – Sistema web*

#### **4.3.3. Interfaces con el software**

Cabe señalar para el uso de este sistema no es necesario ningún tipo de programa adicional, ya que será una plataforma que visualizará gráficos en la web, de modo que el sistema web pedirá los datos que están arrojando los sensores de forma física y son almacenados en la base de datos destinada para este proyecto. El lenguaje de programación adoptado para el desarrollo del sistema web es React JS, el cual es un lenguaje de código abierto y el principal objetivo es la implementación de sitios en una sola página, sin la necesidad de cargar datos nuevamente. Uno de los beneficios de este lenguaje es la creación de aplicaciones que cambian a cada momento, como es el caso del sistema a construir.

#### 4.3.4. Interfaces de comunicaciones

La interfaz de comunicaciones destinada a entregar información acerca de los procesos y herramientas de control a través de lo que el usuario observa habitualmente en la pantalla, utilizará el protocolo TCP / IP que está orientado del cómo se mueve la información desde el remitente hasta el destinatario. Además del uso de una interfaz de programación que consiste principalmente en capturar los datos que se están arrojando el modulo wifi y que son enviado a la base de datos del sistema.

#### 4.4.Requerimientos específicos

A continuación, se especifican los requerimientos del sistema Cyber-Físico del monitoreo en tiempo real de entradas de salida de vehículos en la Universidad del Bío-Bío, principalmente se abortará los requisitos del sistema web.

##### 4.4.1. Requerimientos funcionales del sistema web

**Tabla n° 10:** Requerimientos funcionales del sistema web

<b>ID</b>	<b>Nombre</b>	<b>Descripción</b>
<b>RF_01</b>	Seleccionar fecha	El sistema Cyber-Físico deberá permitir a los usuarios seleccionar una fecha en particular y se despliega el gráfico correspondiente.
<b>RF_02</b>	Visualizar gráficos	El sistema Cyber-Físico deberá permitir a los usuarios visualizar gráficos del monitoreo de vehículos que entran y salen de la Universidad del Bío-Bío.

<b>RF_03</b>	Filtrar gráficos	El sistema Cyber-Físico deberá permitir a los usuarios filtrar gráficos ya sean solo autos que ingresaron, salieron o ambos dos.
--------------	------------------	--

*Fuente: Elaboración propia.*

#### 4.4.2. Interfaces externas de entrada

**Tabla n° 11:** Interfaces externas de entrada

<b>ID</b>	<b>Nombre</b>	<b>Descripción</b>
<b>IE_01</b>	Ingreso al sistema	Se ingresa mediante teclado a la dirección destinada para sistema web ( <a href="http://190.121.25.248:8000/~rodrigo/tesis/app/">http://190.121.25.248:8000/~rodrigo/tesis/app/</a> ).
<b>IE_02</b>	Datos de vehículos entrantes	Información de la base de datos que corresponden a los autos que ingresaron a la UBB.(ID, ID_DEV, TS)
<b>IE_03</b>	Datos de vehículos salientes	Información de la base de datos que corresponden a los autos que salieron de la UBB. (ID, ID_DEV, TS)
<b>IE_04</b>	Datos de vehículos generales	Información de la base de datos que corresponden a los autos en general (ID, NOMBRE, TIPO=0 Entrada de vehículos, TIPO=1 Salida de vehículos, UBICACIÓN)

*Fuente: Elaboración propia.*

#### 4.4.3. Interfaces externas de salida

**Tabla n° 12:** Interfaces externas de salida

<b>ID</b>	<b>Nombre del ítem</b>	<b>Descripción</b>	<b>Medio de salida</b>
<b>IS_01</b>	Descarga de gráficos.	Posibilidad de exportar los gráficos en formato PNG.	PNG por Pantalla

*Fuente: Elaboración propia.*

#### 4.5. Factibilidad del proyecto

El estudio de factibilidad permite analizar la viabilidad del desarrollo del sistema Cyber-Físico propuesto.

Consta de tres aspectos fundamentales:

- **Factibilidad técnica:** Evalúa los recursos de software, hardware y recursos humanos competentes y necesarios para el correcto desarrollo del proyecto.
- **Factibilidad operacional:** Evalúa la viabilidad en cuanto al futuro uso y aceptación de los usuarios finales.
- **Factibilidad económica:** Evalúa la viabilidad en cuanto a los costos del proyecto, es decir, durante el desarrollo y la puesta en marcha. Además, se evalúan los beneficios futuros que se obtendrá al poner en marcha el sistema.

##### 4.5.1. Factibilidad técnica

Este estudio determinará si el equipamiento y los recursos humanos con el que se dispone durante el desarrollo permiten la realización del proyecto.

### Requisitos técnicos para el desarrollo

- Se requiere para la elaboración del sistema web ciertos aspectos técnicos, de los cuales se pueden ver la tabla n° 13.

**Tabla n° 13:** *Requerimientos técnicos del sistema web*

<b>Atributo</b>	<b>Descripción</b>
Lenguaje de programación	Javascript - React JS - PHP - CSS
Sistema gestor de base de datos	MySQL
Lenguaje de consulta	SQL

*Fuente: Elaboración propia.*

**Tabla n° 14:** *Requerimientos técnicos de la programación del IDE-Arduino.*

<b>Atributo</b>	<b>Descripción</b>
Lenguaje de programación	IDE Arduino (Basado en Java)
Funciones principales	Setup() y loop()
Licencia	LGPL o GPL
Sistema operativo	Windows, macOS, Linux

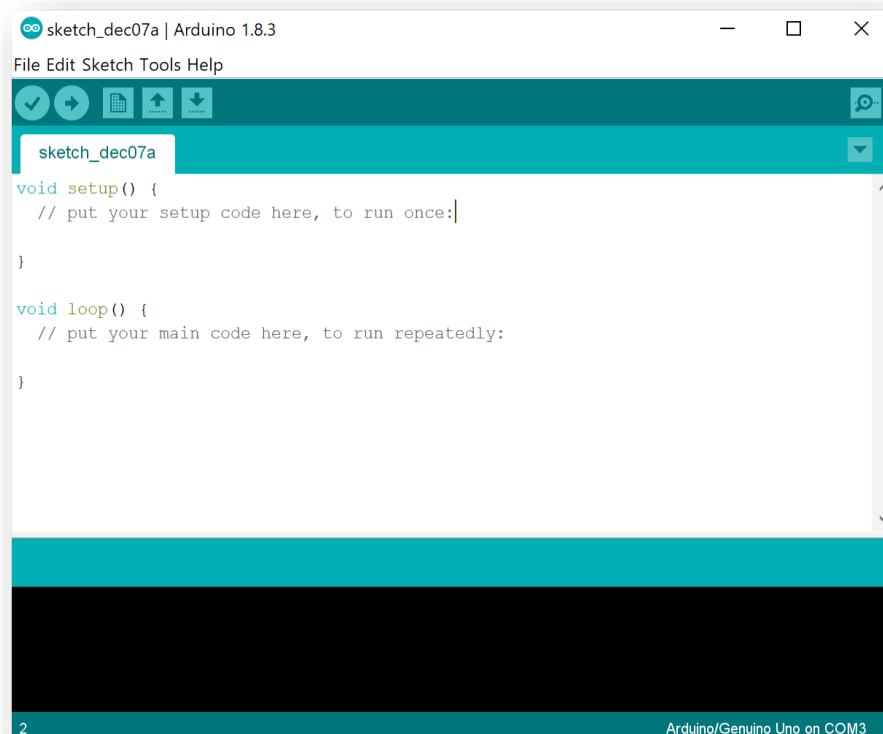
*Fuente: Elaboración propia.*

- **IDE-Arduino**

Aplicación multiplataforma que está escrita en el lenguaje de programación java y su código fuente para el IDE está bajo la licencia pública general de GNU.

La siguiente figura muestra el sketch de visualización del software IDE-Arduino:

**Figura n° 20:** Sketch de visualización Software IDE-Arduino



```

sketch_dec07a
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}

2 Arduino/Genuino Uno on COM3

```

**Fuente:** <https://support.office.com/es-es/article/cargar-c%C3%B3digo-de-panel-y-arduino-ide-a9723765-1314-49e0-a69b-bb5c3e1f628d>

Es completamente factible el desarrollo del sistema Cyber-Físico, dado que se requieren para el desarrollo de software libres y solo implica un gasto en arduinos, sensores y microcontroladores. Se hará uso de un equipo computacional para el desarrollo del sistema web y la programación asociada al arduino y sus componentes, bajo el IDE que fue mencionado anteriormente.

#### 4.5.2. Factibilidad operacional

La factibilidad operacional es la relación con el grado de aceptación que tendrá la solución por parte de los usuarios, quienes serán principalmente funcionarios y administrativos de la Universidad del Bío-Bío, así como los obstáculos que pueden existir para su desarrollo y posterior utilización. Como fue señalado anteriormente, hoy en día la Universidad del Bío-



Bío solo cuenta con un sistema de ingreso / salida por medio de barreras, lo que conlleva a un beneficio si se implementa el sistema Cyber-Físico.

#### 4.5.3. Factibilidad económica

En este último estudio se determinan los recursos necesarios para desarrollar el sistema, los costos en los que se debe incurrir para su fabricación y los beneficios que se obtendrán a partir de la implementación del sistema.

Los principales costos para el desarrollo del sistema son:

- **Hardware** (*componentes de desarrollo arduino*): Los costos asociados estarán dedicados en los materiales ocupados para el desarrollo físico, dado que hubo un gasto de \$18.540.

Dentro de los materiales que fueron necesarios son:

- 2 Arduino micro y accesorios cables.

*Fueron adquiridos en el laboratorio de especialidad que cuenta el Departamento de Sistemas de Información de la Facultad de Ciencias Empresariales.*

- Placa Node MCU Lolin v3: \$ 3.400 c/u. **Total \$6.800.-**

**Cotización:**

*[https://conce.depaquete.cl/index.php?route=product/product&path=59\\_61&product\\_id=53](https://conce.depaquete.cl/index.php?route=product/product&path=59_61&product_id=53)*

- Conversor 3.3V / 5V: \$1.500 c/u. **Total \$3.000.-**

**Cotización:**

*[http://www.arduconce.cl/index.php?route=product/product&product\\_id=190&search=conversor](http://www.arduconce.cl/index.php?route=product/product&product_id=190&search=conversor)*

- 2 sensores ultrasonido: \$3.700 c/u. **Total \$7.400.-**

**Cotización:**

[http://www.arduconce.cl/index.php?route=product/product&product\\_id=78  
&search=sensor+ultra](http://www.arduconce.cl/index.php?route=product/product&product_id=78&search=sensor+ultra)

- Implementación de Placa PCB: 5 Placas. **\$12.534**

**Cotización:** [https://www.mcielectronics.cl/page/fabricacion-pcbs#iframe\\_1](https://www.mcielectronics.cl/page/fabricacion-pcbs#iframe_1)

- **Software:** No fue necesario invertir en nuevo software, dado que todos los software utilizados son de licencia gratuita.

*IDE-Arduino:* <https://www.arduino.cc/en/main/software>

- **Servidor:** No fue necesario invertir en servidor para el sistema web, dado que se facilitó uno por parte de Telefónica del Sur para el desarrollo de este sistema, sin necesidad de invertir en uno propio.

**Tabla n° 15:** Evaluación económica

<b>Evaluación Económica</b>	
Node MCU Lolin v3	<b>\$6.800</b>
Convertor 3.3V / 5V	<b>\$3.000</b>
Sensores Ultrasonido	<b>\$7.400</b>
Implementación Placa PCB	<b>\$12.534</b>

*Fuente:* Elaboración propia.

**Evaluación económica total: \$29.734.-**

**Costos de recursos humanos**

- Los costos de recursos humanos son estimaciones debido a que este proyecto es completamente gratis para el uso de administrativos y funcionarios de la Universidad del Bío-Bío.

**Tabla n° 16:** Costos recursos humanos

<b>Costos recursos humanos</b>	
HH Análisis	<b>300 HH</b>
HH Construcción	<b>150 HH</b>
HH Diseño	<b>16 HH</b>

*Fuente: Elaboración propia.*

**Valor Programador:** \$2.500.- hora.

**Total Horas:** 466 HH.

**Total:** \$1.165.000.-

*HH: Horas hombres*

Es completamente factible el poder desarrollar el proyecto Cyber-Físico, puesto que solo se requiere el costo de implementos electrónicos y su valor no supera gastos mayores.

# **CAPITULO V IMPLEMENTACIÓN**

## 5. Implementación

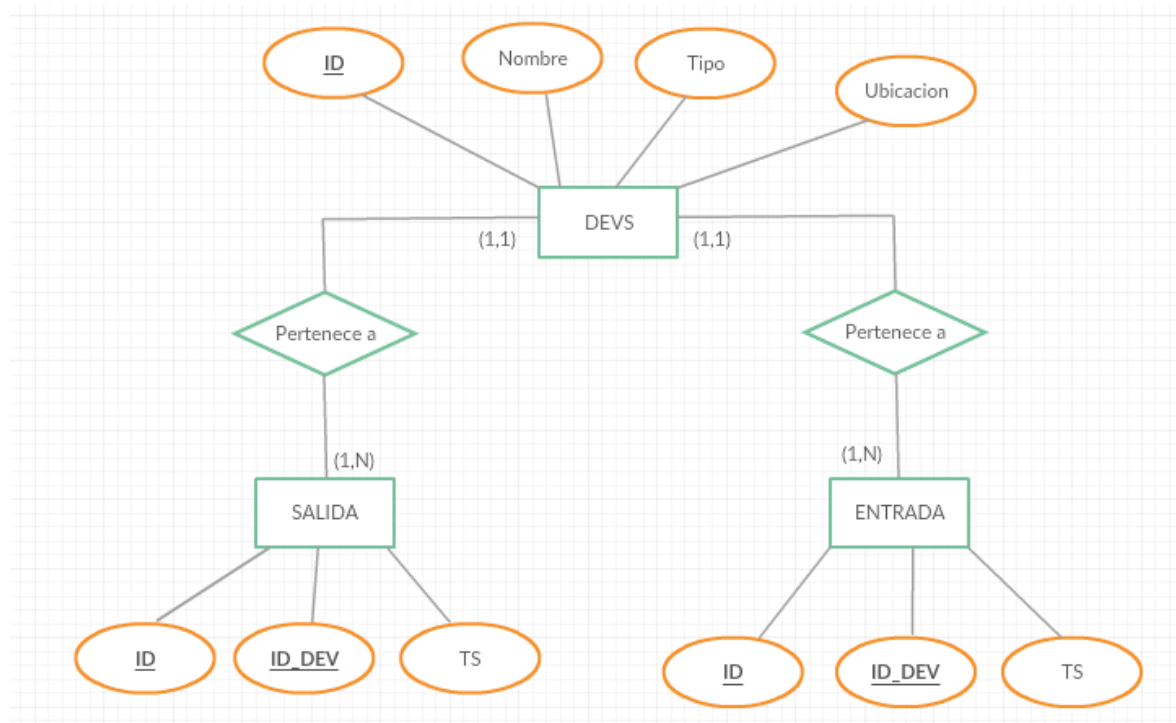
A continuación, se presenta el diseño que rige el desarrollo del Sistema Cyber-Físico. En particular, en este capítulo se aborda la implementación completa que tendrá el sistema en términos estructurales, enfocadas principalmente al sistema web y físico.

### 5.1. Diseño

#### 5.1.1. Modelo entidad Relación

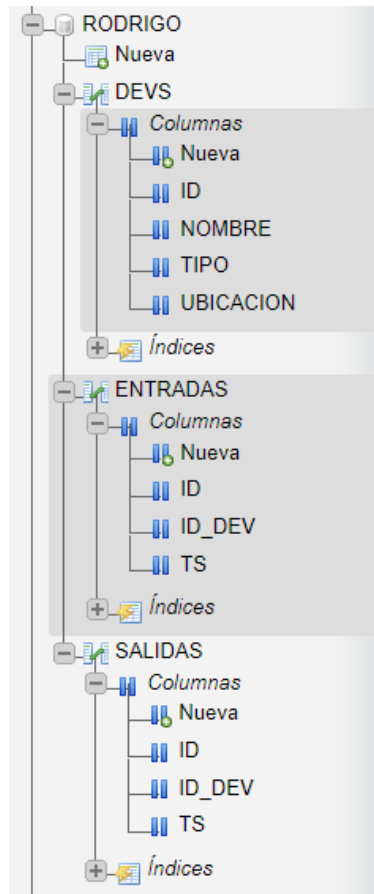
A continuación, se presenta el Modelo Entidad Relación a ocupar para el desarrollo de este sistema. Dentro de ella encontramos tres entidades, las cuales corresponden a DEVS, ENTRADAS y SALIDAS.

**Figura n° 21:** Modelo entidad relación ocupado para el sistema



**Fuente:** Elaboración propia – MER para el sistema web.

**Figura n° 22:** Base de datos a ocupar en servidor.



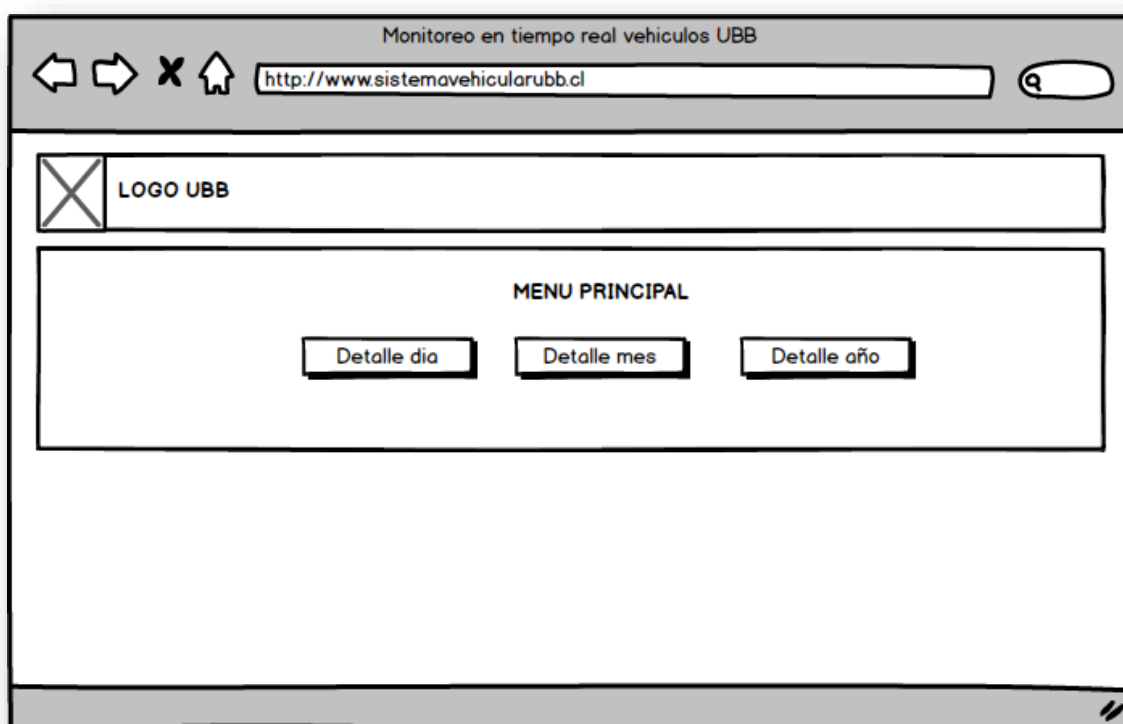
**Fuente:** Elaboración propia – Base de datos Sistema web.

### Diseño de la interfaz (Sistema web)

A continuación, se muestran el diseño de los mockups a implementar en el sistema web que visualizará el usuario, en donde se detallan tres gráficos, ya sea por detalle de día, mes y año.

- En esta interfaz el usuario puede acceder a qué tipo de gráfico desea visualizar, ya sea por día, mes y año.

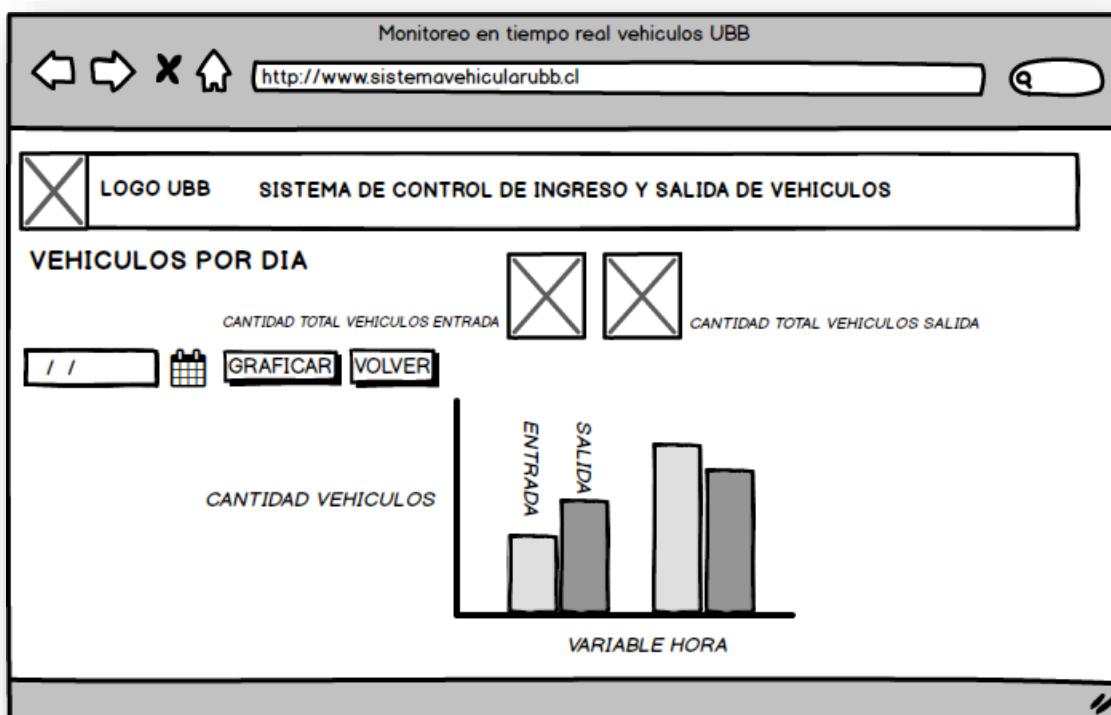
*Figura n° 23: Interfaz para acceso a gráficos.*



*Fuente: Elaboración propia – Mockup.*

- Visualización del gráfico 1 corresponde al ingreso de vehículos dentro de la Universidad del Bío-Bío en un horario determinado. Aquí se debe seleccionar la fecha que desea visualizar y luego se despliega el gráfico seleccionado.

*Figura n° 24: Gráfico 1, corresponde a vehículos / día.*

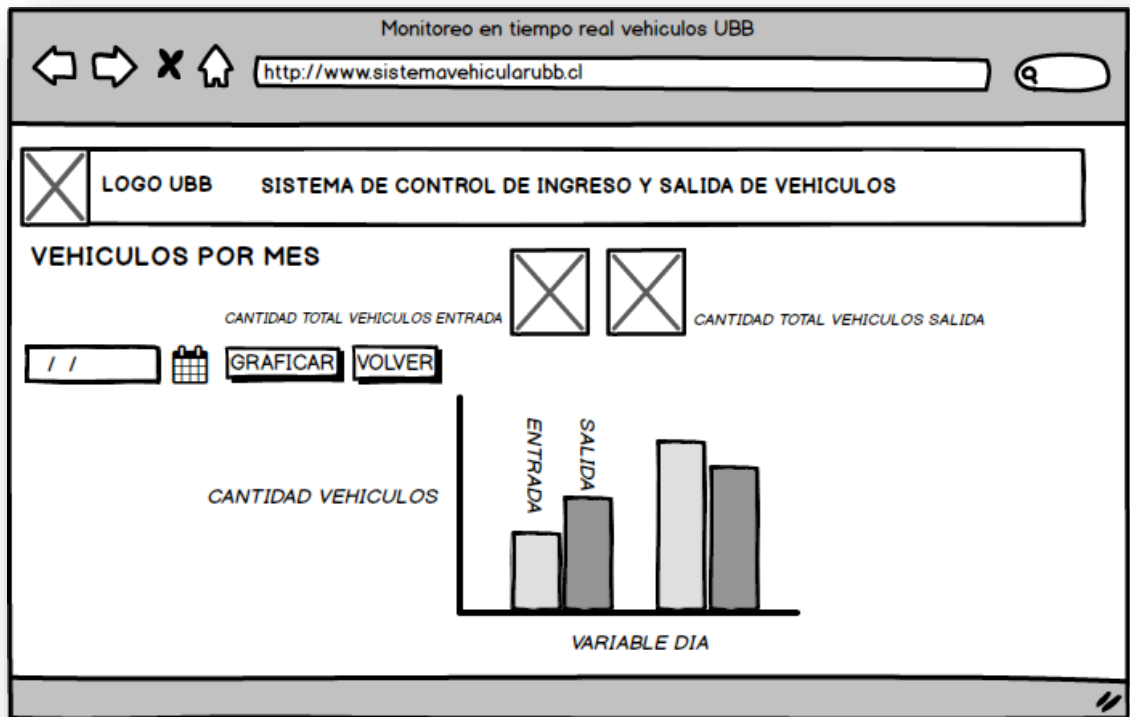


*Fuente: Elaboración propia – Mockup.*



- Visualización del gráfico 2 que corresponde al ingreso de vehículos dentro de la Universidad del Bío-Bío en un determinado mes. Acá se debe seleccionar la fecha que desea visualizar y luego de despliega el gráfico seleccionado.

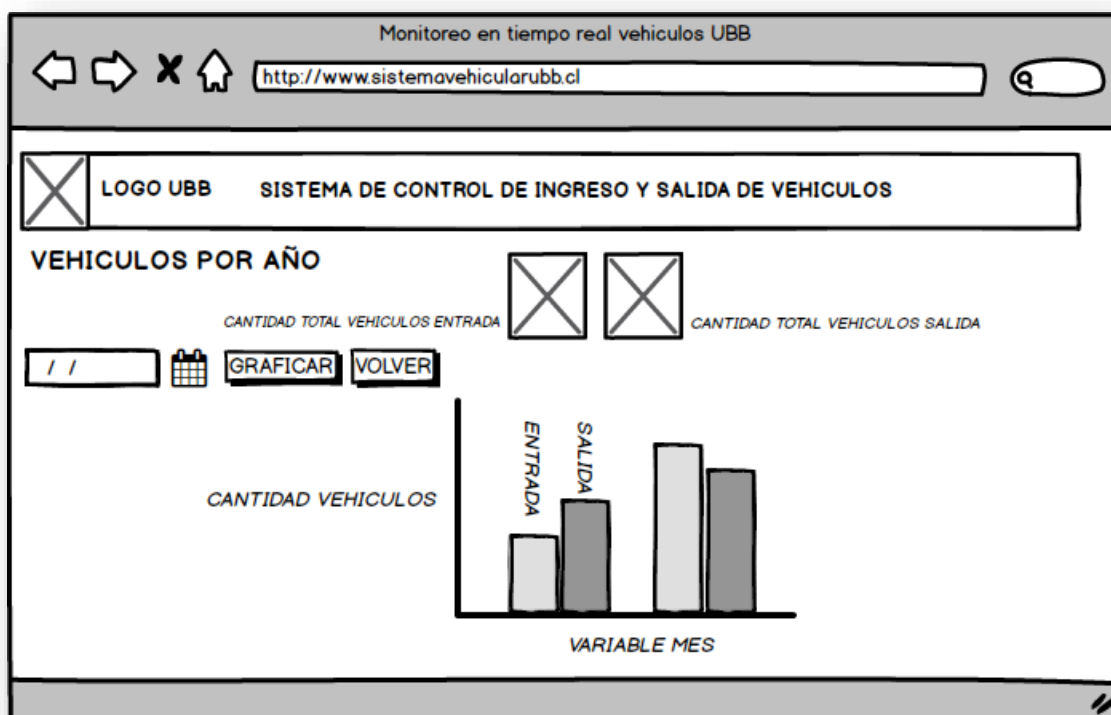
*Figura n° 25: Gráfico 2, corresponde a vehículos / mes.*



*Fuente: Elaboración propia – Mockup.*

- Visualización del gráfico 3 que corresponde al ingreso de vehículos dentro de la Universidad del Bío-Bío en un año determinado. Acá se debe seleccionar la fecha que desea visualizar y luego de despliega el gráfico seleccionado.

*Figura n° 26: Gráfico 3, corresponde a vehículos / año.*



*Fuente: Elaboración propia – Mockup.*

# **CAPITULO VI PRUEBAS Y RESULTADOS**

## 6. Pruebas y Resultados

En esta sección se muestran la prueba realizada para evaluar el funcionamiento del sistema Cyber-Físico.

### Prueba

- La prueba realizada consistió en graficar los vehículos que ingresaron y salieron de la Universidad del Bío-Bío en los días 17 y 18 del mes de enero 2019, estos datos se visualizaron en el sub menú que corresponde al segundo grafico “detalle por mes”, los datos fueron alojados en la base de datos que se dispuso en el servidor virtual (*llamada por nombre BD Rodrigo*). De tal manera se pudiera visualizar los vehículos que entraron y salieron en el mes correspondiente.

**Tabla n° 17:** Detalles de prueba realizada correspondiente al mes de enero 2019

<b>Detalle de la prueba realizada</b>	
<b>Fecha</b>	enero 2019
<b>Días</b>	17/01/2019 18/01/2019
<b>Sensor ultrasonido</b>	Entrada
<b>Sensor ultrasonido</b>	Salida

*Fuente: Elaboración propia*

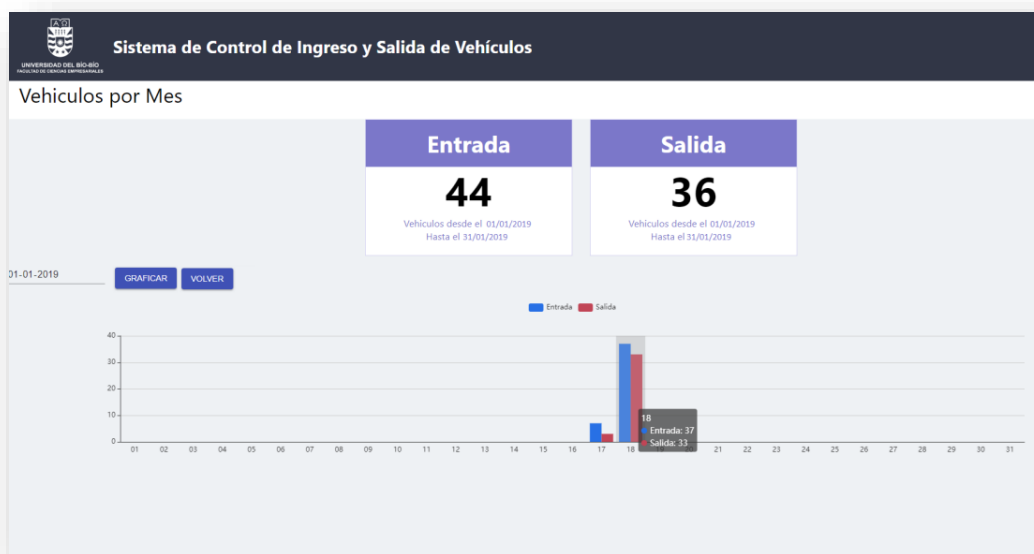
**Resultados obtenidos:**

**Tabla n° 18:** Resultados obtenidos correspondiente al mes de enero 2019

<b>Resultados</b>	
<b>Cantidad de vehículos entrada (17/01/2019)</b>	7
<b>Cantidad de vehículos salida (17/01/2019)</b>	3
<b>Cantidad de vehículos entrada (18/01/2019)</b>	37
<b>Cantidad de vehículos salida (18/01/2019)</b>	33
<b>Total de vehículos entrada (Enero 2019)</b>	44
<b>Total de vehículos salida (Enero 2019)</b>	36

*Fuente: Elaboración propia.*

**Figura n° 27:** Gráfico del sistema web correspondiente al mes de enero 2019



*Fuente: Elaboración propia – Sistema web.*

- De acuerdo a la imagen anterior se visualiza el sistema web en su gráfico de ingreso y salida de los vehículos del mes de enero, cabe señalar que acá se marca la diferencia en el día exacto en el cual hicieron ingreso / salida, esto corresponde a los días 17 y 18 de enero en el cual se realizó la prueba.
- El color azul indica los vehículos que ingresaron y el color rojo los vehículos que salieron de la Universidad del Bío-Bío.
- Se visualiza el total de los vehículos que ingresaron en el mes de enero, un total de 44 que entraron y 36 que salieron.

## **CAPITULO VII CONCLUSIONES**

## 7. Conclusiones

Los objetivos del proyecto se cumplieron en su totalidad, tras un extenso periodo que incluyo la interiorización del manejo de microcontroladores, que destacan principalmente por su bajo costo, multiplataforma y que sean de código abierto, lo que ayudo a la hora de implementar el sistema físico. Así mismo el haber trabajado en proyectos con arduinos en electivo de especialidad que dicta las carreras de informática en la Universidad del Bío-Bío. En relación al sistema web, se complementó a lo ya elaborado y bajo el lenguaje React JS se pudo crear el sistema utilizando una interfaz de programación y así visualizar de manera gráfica las estadísticas del flujo vehicular en la Universidad del Bío-Bío.

En cuanto a la metodología de desarrollo del sistema utilizado, esta se ajustó a lo que se quería implementar en el proyecto, permitiendo generar incrementos en los plazos que fueron establecidos al momento de entregar la propuesta de desarrollo de sistema Cyber-Físico.

Es importante indicar el valor agregado que se obtuvo producto del desarrollo del proyecto, obteniendo una valiosa experiencia, tanto en la motivación de aprender y el conocimiento adquirido durante el desarrollo del sistema, esto me llevo a consolidar los conocimientos entregados por el docente Carlos Araneda, quien dicta electivo de especialidad del área de arduinos, y Patricio Galdames quien fue mi profesor guía, quien tuvo el tiempo necesario y siempre la disponibilidad de solucionar mis inquietudes.

En términos de posibles trabajos futuros como fue señalado anteriormente, la idea es integrar el sistema en una etapa posterior, con el actual sistema de barreras que cuenta la Universidad del Bío-Bío, permitiendo la implementación de un estacionamiento inteligente, que así ayude a los usuarios a tener mayor seguridad y evitar el gran flujo vehicular cuando ingresan y salen de la Universidad del Bío-Bío.

Así mismo cubrir todas las entradas que dispone la Universidad del Bío-Bío, teniendo un total control de vehículos al interior del campus, y así saber a base de estadísticas cual es el ingreso de mayor flujo de vehículos, para luego tomar los determinados planes de contingencia para evitar la saturación del servicio de estacionamientos entregado.



## **Bibliografía**

Rafael Enríquez Herrador (13 de noviembre de 2009). *Guía de Usuario de Arduino*. [http://www.uco.es/aulasoftwarelibre/wpcontent/uploads/2010/05/Arduino\\_user\\_manual\\_es.pdf](http://www.uco.es/aulasoftwarelibre/wpcontent/uploads/2010/05/Arduino_user_manual_es.pdf)

Dani No (2017). *Practica 11: Conexión del procesador ESP8266 como Cliente de un Servidor Web*. <http://www.esploradores.com/conexion-del-procesador-esp8266-como-cliente-de-un-servidor-web/>

Brian W. Evans (Edición española 2011) *Arduino Programming Notebook*. [http://www.ardumania.es/wpcontent/uploads/2011/10/Arduino\\_programing\\_notebook\\_ES.pdf](http://www.ardumania.es/wpcontent/uploads/2011/10/Arduino_programing_notebook_ES.pdf)

*Programando Node MCU, un microcontrolador basado en el ESP8266*  
<https://www.prometec.net/nodemcu-arduino-ide/>

IDE-Arduino (2019). *Web oficial Arduino.cc*. <https://www.arduino.cc/en/main/software>

React js. *Web Oficial React JS* <https://reactjs.org/>

Wikipedia. *Arduino*. <https://en.wikipedia.org/wiki/Arduino#Software>

Web Robótica Arduino micro. *Arduino pro Micro* <https://www.web-robotica.com/arduino/placas-arduino/arduino-micro-genuino-micro-caracteristicas-tecnicas>

Arduino web <https://arduino.cl/que-es-arduino/>

## Anexos

### Código fuente IDE-Arduino

```

#include <SoftwareSerial.h>
#include <NewPing.h>

#define TRIGGER_PIN 14
#define ECHO_PIN 15
#define MAX_DISTANCE 200
#define MIN_DETECT 30
#define MAX_DETECT 150

NewPing sonar(TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE);
SoftwareSerial mySerial(10, 16); // RX, TX

int getDistCM(){
  unsigned long echoTime = sonar.ping_median(10);
  return sonar.convert_cm(echoTime);
}

void setup() {
  // put your setup code here, to run once:
  mySerial.begin(9600);
  mySerial.println("Hello, world?");
}

void loop() {
  int dist;
  unsigned long t0 = millis();
  do{
    dist = getDistCM();
    Serial.print("DIST:");Serial.println(dist);
  }while( dist > MIN_DETECT && dist < MAX_DETECT);

  if( millis()-t0>=5000){
    Serial.println(":DETECT:");
    mySerial.print(":DETECT:");
  }
}

```

**Código fuente ESP8266**

```

#include <ESP8266WiFi.h>
const char *ssid = "iPhone 7 Rodrigo";
const char *password = "redwifi2018";
WiFiClient client;

void doRequest(){
  Serial.println("connecting...");

  // if you get a connection, report back via serial:
  if (client.connect("190.121.25.248", 8000)){
    Serial.println("connected");
    // Make a HTTP request:
    client.println("GET ~/rodrigo/tesis/api.php?cmd=registro&devID=2 HTTP/1.1");
    client.println("Host: 190.121.25.248");
    client.println("Connection: close");
    client.println();
  }
  else{
    // kf you didn't get a connection to the server:
    Serial.println("connection failed");
  }
  delay(500);
  while (client.available()){
    char c = client.read();
    Serial.print(c);
  }

  // if the server's disconnected, stop the client:
  if (!client.connected()){
    Serial.println();
    Serial.println("disconnecting.");
    client.stop();
  }
}

void setup()
{
  Serial.begin(9600);
  delay(10);
}

```

```

// We start by connecting to a WiFi network

Serial.println();
Serial.println();
Serial.print("Connecting to ");
Serial.println(ssid);

/* Explicitly set the ESP8266 to be a WiFi-client, otherwise, it by default,
   would try to act as both a client and an access-point and could cause
   network-issues with your other WiFi-devices on your WiFi-network. */
WiFi.mode(WIFI_STA);
WiFi.begin(ssid, password);

while (WiFi.status() != WL_CONNECTED)
{
  delay(500);
  Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");
Serial.println("IP address: ");
Serial.println(WiFi.localIP());

// give the Ethernet shield a second to initialize:
delay(1000);
}
void loop(){

char buffer[10];
int total = 0;

if( Serial.available() ){
  total = Serial.readBytes(buffer, 8);
  Serial.println(buffer);
  if (buffer[0]=='.' && buffer[1]=='D' && buffer[2]=='E' && buffer[3]=='T' && buffer[4]=='E' && buffer[5]=='C' &&
buffer[6]=='T' && buffer[7]==':'){
    doRequest();
  }
}
}
}

```

## Código fuente Sistema web

```

import React, { Component } from 'react';
import logoUBB from './logo-face-blanco.png';
import Button from '@material-ui/core/Button';
import ReactEcharts from 'echarts-for-react'; // or var ReactEcharts = require('echarts-for-react');
import TextField from '@material-ui/core/TextField';
import './App.css';

var api_url = 'http://190.121.25.248:8000/~rodrigo/tesis/api.php';

class Detalle extends Component {

  constructor(props) {
    super(props);
    this.state = {
      selectedDate: '2019-01-01',
      dataLabel: [],
      dataS1: [],
      dataS2: [],
      titulo:"",
      totalEntrada:0,
      totalSalida:0,
      fechaInicio:'-',
      fechaFin:"",
    }
  }

  componentDidMount() {
    console.log("CARGA DATOS PARA " + this.props.tipo);

    if (this.props.tipo === 'dia') {
      this.setState({ titulo:'Vehiculos por Día'})
    }
    if (this.props.tipo === 'mes') {
      this.setState({ titulo:'Vehiculos por Mes'})
    }
    if (this.props.tipo === 'año') {
      this.setState({ titulo:'Vehiculos por Año'})
    }
  }
}

```

```

handleDateChange(date) {
  this.setState({ selectedDate: date.target.value });
}

async updateGraph() {
  var matches = this.state.selectedDate.match(/([0-9]+)-([0-9]+)-([0-9]+)/);
  if (!matches) {
    return;
  }
  let year = parseInt(matches[1]);
  let month = parseInt(matches[2]);
  let day = parseInt(matches[3]);
  let fechaInicio = `${day}/${month}/${year}`
  let fechaFin = `${day}/${month}/${year}`
  let diasMeses = [31,28,31,30,31,30,31,31,30,31,30,31]
  let url = "";
  if (this.props.tipo === 'dia') {
    url = `${api_url}?cmd=consulta&tipo=dia&dia=${day}&mes=${month}&year=${year}`;
    fechaInicio = `${day}/${month}/${year}`;
    fechaFin = `${day}/${month}/${year}`
  }
  else if (this.props.tipo === 'mes') {
    url = `${api_url}?cmd=consulta&tipo=mes&mes=${month}&year=${year}`;
    fechaInicio = `01/${month}/${year}`;
    fechaFin = `${diasMeses[month-1]}/${month}/${year}`
  }
  else if (this.props.tipo === 'año') {
    url = `${api_url}?cmd=consulta&tipo=year&year=${year}`;
    fechaInicio = `01/01/${year}`;
    fechaFin = `31/12/${year}`
  }
  let r1 = await fetch(url)
  let r2 = await r1.json();

  console.log("UPDATE");
  console.log(url);
  console.log(r2);

  let dataLabel = r2.label;
  let dataS1 = r2.s1;
  let dataS2 = r2.s2;

```

```

let totalS1 = dataS1.reduceRight(function(a,b){return a+b;})
let totalS2 = dataS2.reduceRight(function(a,b){return a+b;})

this.setState({ dataLabel: dataLabel, dataS1: dataS1, dataS2: dataS2, totalEntrada:totalS1, totalSalida:totalS2,
fechaInicio:fechaInicio, fechaFin:fechaFin});
}

render() {
let option = {
color: ['#2770e5', '#c14556'],
tooltip: {
trigger: 'axis',
axisPointer: {
type: 'shadow'
}
},
legend: {
data: ['Entrada', 'Salida']
},
toolbox: {
show: true,
orient: 'vertical',
left: 'right',
top: 'center',
feature: {
mark: { show: true },
dataView: { show: true, readOnly: false },
magicType: { show: true, type: ['line', 'bar', 'stack', 'tiled'] },
restore: { show: true },
saveAsImage: { show: true }
}
},
calculable: true,
xAxis: [
{
type: 'category',
axisTick: { show: false },
data: this.state.dataLabel
}
],

```

```

yAxis: [
  {
    type: 'value'
  }
],
series: [
  {
    name: 'Entrada',
    type: 'bar',
    barGap: 0,
    data: this.state.dataS1
  },
  {
    name: 'Salida',
    type: 'bar',
    data: this.state.dataS2
  },
]
};

console.log(this.state.dataLabel.length);

let chart = null;
if (this.state.dataLabel.length > 0) {
  chart =
  <ReactEcharts
    option={option}
    notMerge={true}
    lazyUpdate={true}
    theme={"theme_name"}
  />
} else {
}

console.log("this.state.dataS1", this.state.dataS1);
console.log("this.state.dataS2", this.state.dataS2);

return (
  <div style={{ display: 'flex', flexDirection: 'column' }}>
    <div className="header1">
      <img alt="logoubb" src={logoUBB} style={{ width: '180px' }}></img>

```



```

    <div className="headerText">Sistema de Control de Ingreso y Salida de Vehículos</div>
  </div>
  <div className="header2"><center>{this.state.titulo}</center></div>
  <div className="content">
    <div><Card1 nombre="Entrada" numero={this.state.totalEntrada} sub1={`Vehiculos desde el
    ${this.state.fechaInicio}`} sub2={`Hasta el ${this.state.fechaFin}`} /></div>
    <div><Card1 nombre="Salida" numero={this.state.totalSalida} sub1={`Vehiculos desde el
    ${this.state.fechaInicio}`} sub2={`Hasta el ${this.state.fechaFin}`} /></div>
  </div>
  <div style={{ display: 'flex', flexDirection: 'row', flex: 1 }}>
    <span><form noValidate> <TextField id="date" label="Fecha" type="date" format={'DD/MM/YYYY'}
    defaultValue={this.state.selectedDate} InputLabelProps={{ shrink: true, }}
    onChange={this.handleChange.bind(this)} /> </form></span>
    <span><Button variant="contained" color="primary" style={{ margin: 20, }} onClick={() => {
    this.updateGraph() }}>Graficar</Button></span>
    <span><Button variant="contained" color="primary" style={{ margin: 20, }} onClick={() => {
    this.props.nav('MainScreen') }}>Volver</Button></span>
  </div>
  <div className="chart">{chart}</div>
</div>
);
}
}

class MainScreen extends Component {

  render() {

    return (
      <div style={{ display: 'flex', flexDirection: 'column' }}>
        <div className="header1">
          <img alt="logoubb" src={logoUBB} style={{ width: '180px' }}></img>
          <div className="headerText">Sistema de Control de Ingreso y Salida de Vehículos</div>
        </div>
        <div className="header2"><center>Menú Principal</center></div>
        <div className="content" style={{ justifyContent: 'center', alignContent: 'center' }}>
          <Button variant="contained" color="primary" style={{ margin: 20, }} onClick={() => { this.props.nav('Detalle',
          { tipo: 'dia' }) }}>DETALLE POR DIA</Button>
          <Button variant="contained" color="primary" style={{ margin: 20, }} onClick={() => { this.props.nav('Detalle',
          { tipo: 'mes' }) }}>DETALLE POR MES</Button>
        </div>
      </div>
    );
  }
}

```

```

    <Button variant="contained" color="primary" style={{ margin: 20, }} onClick={() => { this.props.nav('Detalle',
{ tipo: 'año' }) }}>DETALLE POR AÑO</Button>
  </div>
</div>
);
}
}

class Card1 extends Component {
  render() {
    return (
      <div className="card1">
        <div className="card1Head">
          {this.props.nombre}
        </div>
        <div className="card1Body">
          {this.props.numero}
          <div className="card1BodySubText">{this.props.sub1}</div>
          <div className="card1BodySubText">{this.props.sub2}</div>
        </div>
      </div>
    );
  }
}

class App extends Component {

  constructor(props) {
    super(props);
    this.state = {
      currentPage: <MainScreen nav={ this.nav.bind(this) } />,
    }
  }

  nav(nuevaPagina, props) {

    var comp = [];
    if (nuevaPagina === 'MainScreen') {
      comp = <MainScreen nav={ this.nav.bind(this) } {...props} />
    }
  }
}

```

```

else if (nuevaPagina === 'Detalle') {
  comp = <Detalle nav={this.nav.bind(this)} {...props} />
}
else {
  comp = <MainScreen nav={this.nav.bind(this)} {...props} />
}

this.setState({ currentPage: comp })
}

render() {

  return (<div>{this.state.currentPage}</div>);
}
}

export default App;

```

```

.App {
  text-align: center;
}

.App-logo {
  animation: App-logo-spin infinite 20s linear;
  height: 40vmin;
}

.App-header {
  background-color: #282c34;
  min-height: 100vh;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  font-size: calc(10px + 2vmin);
  color: white;
}

.App-link {
  color: #61dafb;
}

```

```
}  
  
@keyframes App-logo-spin {  
  from {  
    transform: rotate(0deg);  
  }  
  to {  
    transform: rotate(360deg);  
  }  
}  
  
.header1 {  
  background-color: #313646;  
  flex: 1;  
  height: 120px;  
  align-items: center;  
  display: flex;  
  color: #FFFFFF;  
  font-size: calc(20px + 1vmin);  
  font-weight: bold;  
}  
  
.header2 {  
  background-color: #FFFFFF;  
  flex: 1;  
  height: 65px;  
  font-size: calc(25px + 1vmin);  
  padding-left: 20px;  
}  
  
}  
  
.card1 {  
  border: 1px solid #d3d1f2;  
  background-color: #ffffff;  
  width: 350px;  
  height: 230px;  
  margin-left: 30px;  
}  
  
.card1Head {  
  background-color: #7b77c9;  
  height: 80px;  
  color: #FFFFFF;  
  align-items: center;  
  justify-content: center;
```

```

display: flex;
flex-direction: column;
font-size: calc(30px + 1vmin);
font-weight: bold;
}
.card1Body{
color: #000000;
align-items: center;
justify-content: center;
display: flex;
flex-direction: column;
font-size: calc(60px + 1vmin);
font-weight: bold;
height: 120px;
}
.content{
align-items: center;
justify-content: center;
display: flex;
flex-direction: row;
}
.card1BodySubText{
font-size: calc(7px + 1vmin);
font-weight: normal;
color:#7b77c9;
}
.chart{
display: block;
}

body{
background-color: #eef1f4;
}

```