



Anteproyecto

“IoT-Parking”



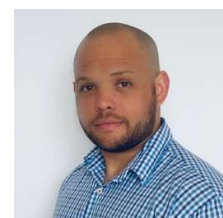
Juan Ignacio Scaffo – 257565

Ing. Electrónica



Carlos Casco – 251942

Ing. Telecomunicaciones



Ronny Suárez – 238874

Ing. Telecomunicaciones

Profesores:

Emiliano Espíndola González

Juan Pedro Silva Gallino

Contenido

I.	Introducción.....	2
II.	Objetivo General	2
III.	Descripción.....	2
i.	Problema Principal	2
ii.	Propuesta	2
iii.	Objetivos Principales.....	2
iv.	Objetivo Adicional	3
v.	Diagrama tentativo	3
vi.	Interacción entre sistemas.....	4
IV.	Casos de Uso	4
i.	Reservar lugar en el estacionamiento	4
ii.	Llegada al estacionamiento	4
V.	Componentes del sistema	5
VI.	Plan de Trabajo.....	7
i.	Mecanismo de barrera	7
ii.	Flujo de datos a la nube	7
iii.	Mostrar lugar para estacionar	8
iv.	Sistema de reserva	8
v.	Ruta hacia el lugar de estacionar	8
VII.	Riesgos	9
VIII.	Prototipos.....	9
i.	Prototipo I.....	9
ii.	Prototipo II.....	9
IX.	Diagrama de GANTT	10
X.	Materiales Requeridos	11
i.	Módulo sensor de proximidad / obstáculos MK0434.....	11
ii.	Microcontrolador ESP32-CAM.....	11
iii.	FTDI: USB TTL FT232RL	11
iv.	Servomotor: Micro Servo SG90	11
v.	Display: OLED 0.96" 128x64	12
vi.	Tira LED: WS2812B 5050	12
XI.	Presupuesto.....	13
XII.	Referencias	13

I. Introducción

El auge y desarrollo de la industria automotriz ha traído muchos beneficios a nuestras vidas ya que el auto nos permite movilizarnos de manera rápida a nuestros centros de trabajo, hogares, centros comerciales, etc.

Hoy día en las principales ciudades de cada país hay un creciente número de autos y esto provoca cierto caos a la hora de conducir en horas pico. Esto trae otro problema real a la mesa y es la dificultad a la hora de estacionar en ciertos lugares como centros comerciales e instituciones de salud donde hay mucho movimiento de personas y en ocasiones perdemos tiempo encontrando un lugar donde dejar nuestro vehículo.

Presentamos IoT-Parking, un sistema de estacionamiento inteligente utilizando las herramientas que nos brinda IoT para detectar posiciones libres en un estacionamiento sin la necesidad de perder tiempo en encontrar un lugar adecuado para estacionar.

II. Objetivo General

Nuestro objetivo es brindar una alternativa a los estacionamientos tradicionales dotando nuestro sistema de inteligencia para que sea eficiente, práctico y atractivo al cliente a la hora de estacionar su vehículo.

III. Descripción

i. Problema Principal

El principal problema que abordaremos en nuestro proyecto es **el tiempo que emplea cada persona para estacionar en lugares concurridos.**

Hoy día queremos siempre “ganar tiempo” y eso nos lleva a tomar decisiones erróneas como estacionar en lugares no habilitados. Esto tiene consecuencias negativas como accidentes, multas y estrés a las personas involucradas. En lugares como hospitales, centros educativos, centros comerciales, centros de negocios, etc. esta situación se acentúa.

ii. Propuesta

Nuestro equipo propone dotar de inteligencia a los estacionamientos, utilizar sensores y procesamiento de datos para resolver de manera eficiente los problemas planteados.

El estacionamiento tendrá información en tiempo real de los lugares disponibles y otorgará al cliente un espacio adecuado para su auto.

Además, puede ser capaz de guiarlo hasta su lugar, reduciendo incluso más las tareas que el cliente debe llevar a cabo.

Pensamos que sería útil incluir en un estacionamiento inteligente un sistema de reservas, así las personas que tienen preferencia por un determinado lugar, puedan gozar del beneficio de estacionar donde gusten.

iii. Objetivos Principales

- Procesar la información de los sensores para otorgarle al usuario su lugar disponible.
- Establecer un reconocimiento de matrícula a partir de una imagen utilizando un OCR (Optical Recognition System).
- Sistema de registro de usuarios, para poder identificar a los clientes utilizando sus matrículas, el sistema debe ser capaz de emitir alertas al usuario en el caso que se identifique su matrícula.
- Sistema de reservas, el usuario debe poder seleccionar un lugar específico, así como fecha, hora y la duración de su parking.
- El movimiento mecánico de las múltiples barreras para

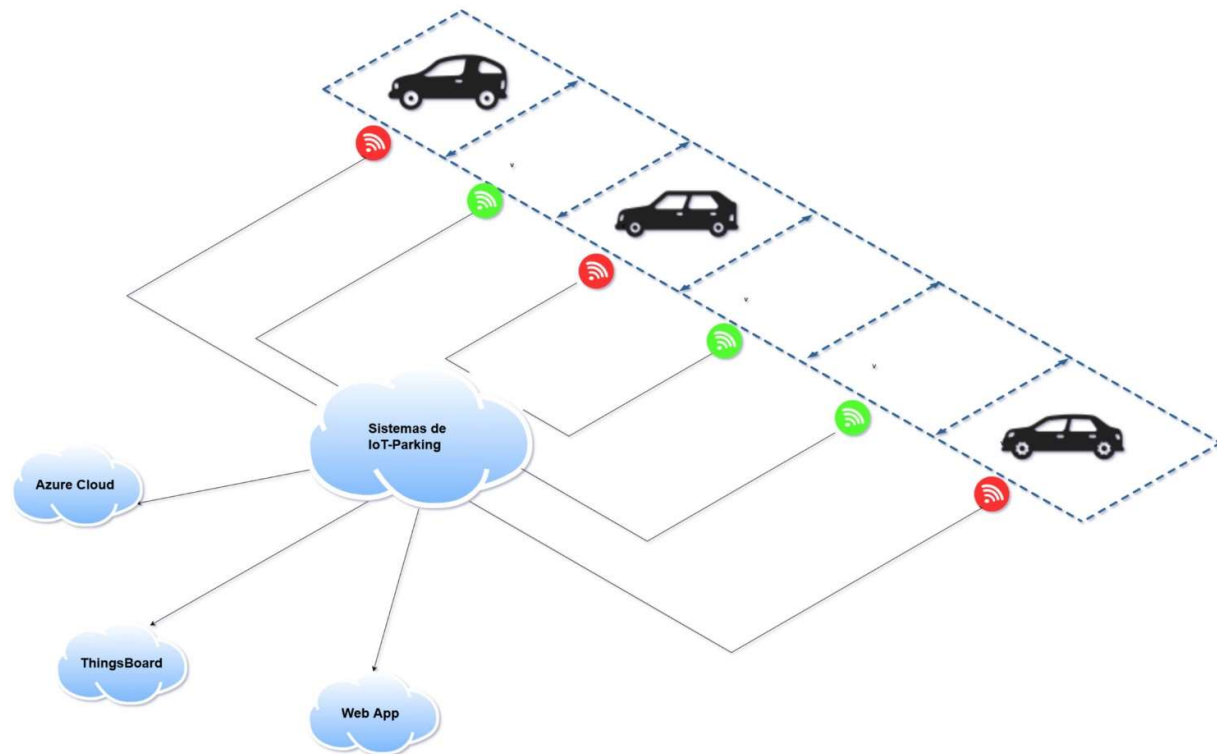
iv. Objetivo Adicional

- Detectar la posición del auto adentro del estacionamiento para lograr guiarlo hasta su lugar utilizando IPS (Indoor Positioning System).

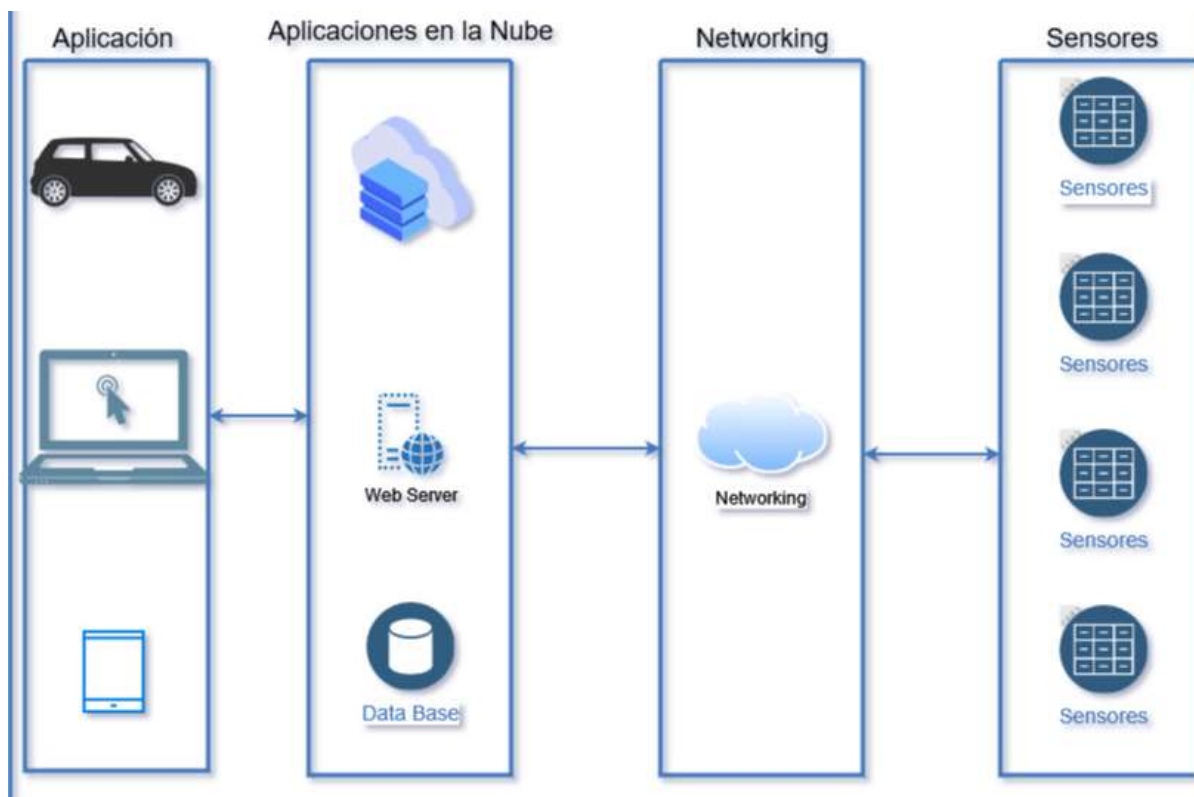
A nivel de aprendizaje, nos interesa profundizar en estos aspectos:

- Procesar los datos en un servidor externo y volver a recibir los datos para procesarlos.
- Aprender a utilizar e implementar REST APIs.

v. Diagrama tentativo



vi. Interacción entre sistemas



IV. Casos de Uso

A continuación, explicaremos algunos casos de uso y el procedimiento que se debe ejecutar en cada caso:

i. Reservar lugar en el estacionamiento

1. El usuario inicia sesión en el sistema de reservas.
2. Se dirige a la sección de "Reservar" donde podrá ver un diagrama con los lugares disponibles.
3. Selecciona un lugar, horario de ingreso y duración estimada.

ii. Llegada al estacionamiento

1. Un auto se presenta en el portón principal del estacionamiento.
2. Se detecta con un sensor su presencia y se toma una captura utilizando una cámara.
3. La imagen es enviada a un servidor que utilizará OCR para descifrar la matrícula del auto.
4. El usuario registrado con esa matrícula es notificado por una web app o aplicación si desea ingresar en el estacionamiento.
5. El usuario notifica que desea ingresar en el estacionamiento.
6. Se abre la barrera del portón principal y se muestra en un display el lugar que le corresponde.
7. El auto ingresa en el estacionamiento y se ilumina un camino de leds hacia el lugar asignado.
8. Como objetivo adicional nos gustaría cumplir que si el auto gira el sistema debe ser capaz de actualizar el camino de leds para guiarlo.

De la descripción de los casos de uso podemos resumir que **las acciones que el usuario puede ser capaz de ejecutar son las siguientes:**

- Registrarse como cliente en la web app.
- Ingresar en el estacionamiento con su auto.

- Poder visualizar el lugar que le fue asignado
- Ser guiado hacia su lugar con un camino de luces.
- Reservar un lugar en el estacionamiento.

V. Componentes del sistema



Imagen 1 - Sensor

Los sensores de posición / proximidad serán indispensables en la etapa inicial para detectar la cercanía del auto al portón principal y en cada lugar del estacionamiento. Para cada lugar del estacionamiento se realizará una estructura similar que se repetirá y se dirigirá a un sistema que codificará cada señal de manera que sea sencillo detectar para cada posición si está ocupado o disponible. Para distribuir las diferentes señales pensamos en utilizar ESP8266 que actúe como un hub de los sensores provenientes de por lo menos 2 lugares de estacionamiento y que cada uno de estos hubs envíe a un ESP32 a través de la nube.



Imagen 2- ESP32

La ESP32 será el microcontrolador que se utilizará en el proyecto. Su integración Wifi/Bluetooth es indispensable para lograr una solución IoT. En particular utilizaremos el ESP32-CAM principalmente por su fácil incorporación de una cámara para nuestro uso.

La cámara es indispensable para el reconocimiento de la matrícula, la idea es que a partir del sensor de posicionamiento el ESP32 envíe una señal a la cámara para tomar una foto. Inmediatamente será subida a un servidor web que validará en una base de datos y se notificará al usuario.



Imagen 3 - Luces LEDs

Se utilizarán tiras led RGB para que se puedan colocar fácilmente en la maqueta y además sean controlables individualmente con el ESP32, a partir de su manejo fácil a través de librerías nos permite un mayor control si se cumple el objetivo adicional de generar una nueva ruta si el auto se desvía en el camino a su lugar disponible.

En la descripción de los materiales, profundizaremos en las tiras led que se utilizarán concretamente.

*Imagen 4 - Servo Motor*

El servo motor es un actuador adecuado para lograr el movimiento de las diferentes barreras de estacionamiento que tendrán que abrirse y cerrarse, no solo la del portón principal, sino que también la de cada lugar del estacionamiento.

*Imagen 5 - Display OLED 0.96"*

El display permitirá informar al usuario cual es el lugar asignado. El equipo ya cuenta en su disposición con una pantalla OLED de 0,96 pulgadas, pero también es posible utilizar un display LCD de 16x2.

La plataforma cloud es uno de los pilares fundamentales del proyecto, se realizaron pruebas de integración entre diferentes tecnologías y se evaluó que Thingsboard puede ser una herramienta ideal para la implementación, principalmente por ser gratuito y la habilidad para responder a eventos a través de las "rule chain" que permiten incluso la comunicación con una REST API y devolverle al dispositivo la información necesaria.



Además, consideramos alojar nuestra Web App, así como el servicio de procesamiento de imágenes utilizando una máquina virtual alojada con el servicio cloud de Microsoft Azure. La elección de utilizar Azure proviene principalmente de la familiaridad con el entorno y sus precios más económicos en comparación con los servicios de otras empresas.

El web server que recibirá la imagen y la procesará será un servidor creado utilizando node.js y SQL Express que procesará la imagen con tesseract.js.

Una vez detectado se validará con un SQL server utilizando el npm package mysql.



VI. Plan de Trabajo

Para desarrollar la planificación del plan de trabajo tuvimos en cuenta diferentes prácticas de gestión de proyectos que nos permiten acercarnos a cumplir los objetivos en un tiempo adecuado.

En primer lugar, nos parece fundamental que las pruebas de concepto sean lo más detalladas posibles en su documentación. Es decir, se realizará un documento para cada prueba de concepto con tres etapas. La primera etapa detallará su objetivo general y sus sub objetivos correspondientes, esta documentación se realizará previo a la construcción de la prueba de concepto y será una continuación de la descripción de cada prueba que se encuentra en este documento.

La segunda etapa se realizará una vez construida y probada la prueba de concepto, se detallarán los diferentes problemas encontrados y también todos los aspectos que ya están funcionales y no necesitan modificación.

La tercera y última etapa consiste en detallar diferentes soluciones a los problemas encontrados en la etapa anterior, se deberá documentar la solución a los problemas y ejecutarlas para poder avanzar a las siguientes pruebas de concepto.

Otro aspecto importante que deben tener las pruebas de concepto es que se prueben conceptos concretos y no se abarquen múltiples conceptos juntos en una sola prueba. Las pruebas de concepto como indican su nombre deben probar exhaustivamente un solo concepto y no varios.

A continuación, se detallarán las diferentes pruebas de concepto planificadas.

i. Mecanismo de barrera



Prueba 1 - Mecanismo de barrera

Esta prueba de concepto inicial tiene como objetivo diseñar y probar el proceso mecánico que se encontrará en el portón principal y en los lugares del estacionamiento. Se deberán probar varios servomotores conectados de manera simultánea y lograr elegir cuál de las barreras se abren.

ii. Flujo de datos a la nube



Prueba 2 - Flujo de datos

En esta prueba de concepto, se deben enviar datos al servidor web, preferencialmente, los datos binarios de la imagen, y recibir una respuesta de este en el ESP32-CAM.

Además, se manejará la verificación del usuario en aceptar abrir el portón o denegarlo.

iii. Mostrar lugar para estacionar



Prueba 3 - Lugar asignado

El propósito general de esta prueba de concepto debe ser el procesamiento de los múltiples datos captados por los sensores de los lugares del estacionamiento, principalmente se deberá calcular a partir de los mismos el próximo lugar para asignar y mostrarlo en el display.

iv. Sistema de reserva



Prueba 4 - Sistema de reserva

El sistema de reservas será una capa adicional que se incorporará por encima del aviso de la asignación de lugares y el flujo de datos. En esta prueba de concepto, es importante que se pueda seleccionar, lugar, fecha y hora desde la web app. Si el auto con la matrícula de la reserva ingresa en el estacionamiento, se debe preguntar al servidor cuál lugar está reservado. Si un auto nuevo ingresa en el estacionamiento, no se le debe otorgar un lugar reservado.

v. Ruta hacia el lugar de estacionar



Prueba 5 - Guía para estacionar

La prueba 5 será la última pero la más desafiante. A partir de una serie de caminos de leds y dos posiciones, la posición del auto que va a cambiar continuamente y la posición del lugar del estacionamiento, se deben iluminar leds particulares que guíen al auto al lugar disponible.

Como se explicó previamente, de manera adicional nos gustaría obtener automáticamente y de manera precisa la posición del auto a pesar de que se mueva a diferentes velocidades y conseguir a partir del mismo los leds necesarios para guiar el auto. Sin embargo, inicialmente en esta prueba de concepto la luz led se prenderá cuando un lugar sea asignado y quedará fija hasta que el auto se estacione.

Para lograr el objetivo adicional toma principal importancia el IPS (Indoor Positioning System) un IPS nos permite encontrar de manera precisa un objeto en un lugar de tamaño reducido que el GPS no cuenta con la precisión para lograr.

VII. Riesgos

Con estas diferentes pruebas de concepto identificamos una serie de riesgos que pueden afectar el desarrollo del mismo, se detallan a continuación:

- Dificultad en la detección de texto, debido a la calidad de la imagen y/o tiempo de subida demasiado largo.
- Dificultad en la manera que se unifican las diferentes señales de los sensores en el ESP32, la solución no escala fácilmente debido a la gran cantidad de pines que se deben utilizar por cada lugar.
- Problemas para el manejar los datos de los sensores y las reservas simultáneamente en el momento de las reservas.

VIII. Prototipos

Los prototipos serán integraciones de varias pruebas de conceptos que nos permitan aproximarnos a la maqueta final.

i. Prototipo I

El prototipo 1 junta las primeras 4 pruebas de concepto. En este prototipo la guía de leds es fija y no cambia en función a la posición del auto.

Objetivos del prototipo:

- Manejar el flujo de datos entre el servidor, Thingsboard y el ESP32.
- Flujo de los datos provenientes de los múltiples sensores y centralizarlos en el ESP32.
- Manejo simultáneo de reservas y lugares disponibles.

ii. Prototipo II

En este prototipo se intentará el cambio de la ruta en la guía de leds, pero a partir de la posición que se enviará manualmente.

Objetivo del prototipo:

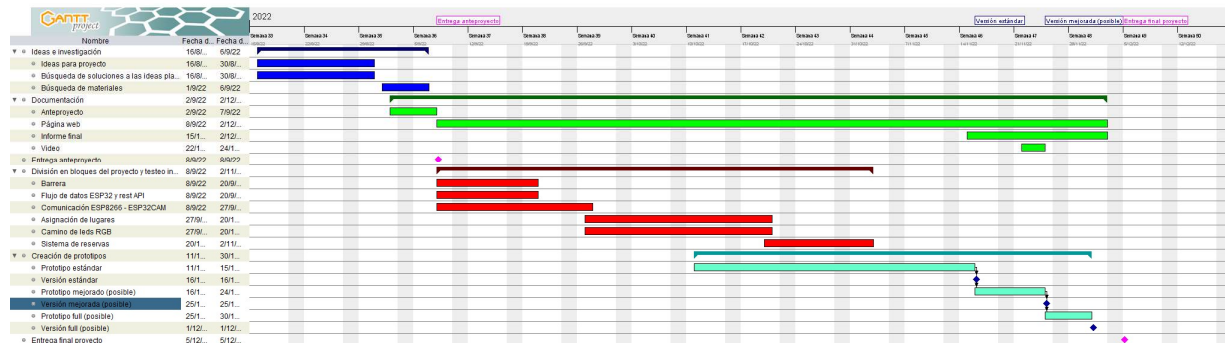
- Cálculo de LEDS que deberían prenderse en el caso que el auto cambie su ruta.

Objetivo adicional:

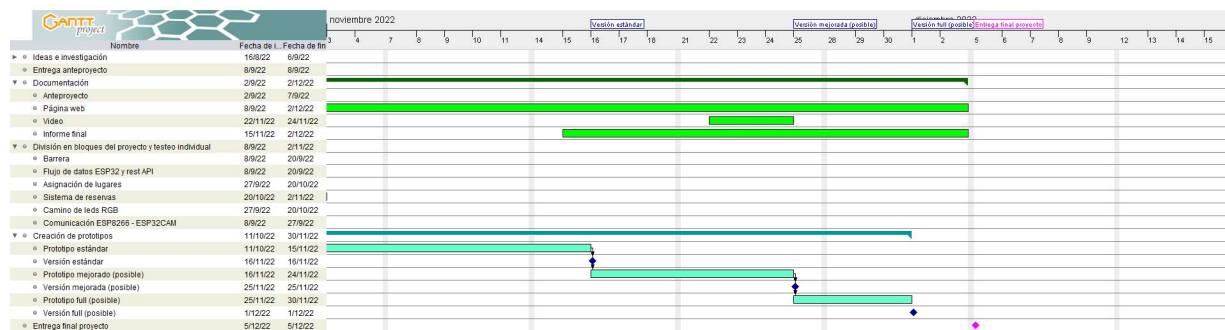
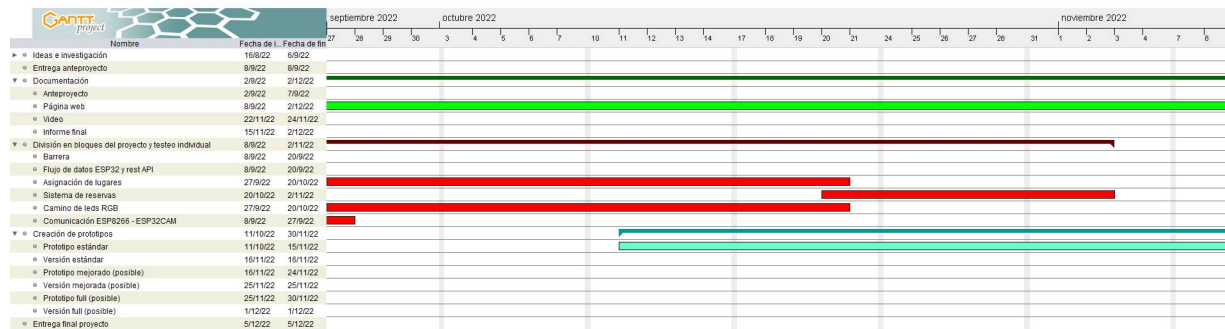
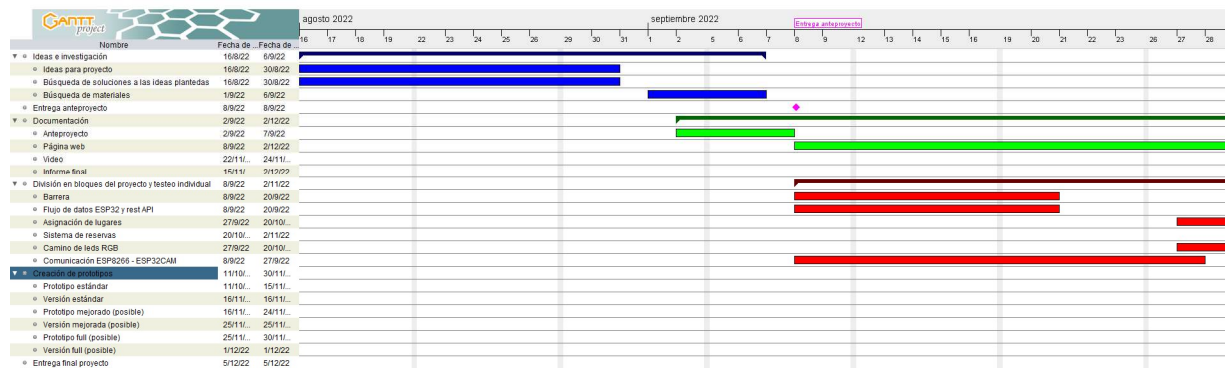
- La detección de la posición a través de un IPS.
- El envío constante de los datos de posición a Thingsboard.

Una vez completados los dos prototipos, el funcionamiento de cada componente ya estará implementado y nos centraremos en comenzar a diseñar la maqueta final.

IX. Diagrama de GANTT



Detallado por partes:



X. Materiales Requeridos

i. Módulo sensor de proximidad / obstáculos MK0434ⁱ

Para los sensores de posición se eligió los infrarrojos principalmente porque resultan más económicos en comparación a otros sensores y su salida digital facilita el manejo de los datos desde el ESP32.

- Detectar la presencia de obstáculos entre 2 a 50 cm
- Angulo de detección 35°
- Comparador LM393 on-board
- Orificio de instalación para facilitar su uso.
- Indicador de alimentación (LED rojo)
- Indicador de salida digital (LED verde)
- conexión de 3 hilos
- Tensión de alimentación: 3.3v a 5v

ii. Microcontrolador ESP32-CAMⁱⁱ

El ESP32-CAM es un microcontrolador que contiene todas las ventajas de un ESP32 normal pero además agrega la posibilidad de agregar una cámara de manera sencilla.

- RAM 520KB SRAM +4M PSRAM
- Interfaces UART, SPI, I2C and PWM
- Cámara de 2 Megapixel
- Peso 10g
- UART Baudrate por defecto 115200

iii. FTDI: USB TTL FT232RLⁱⁱⁱ

El ESP32-CAM no tiene un puerto USB para la programación desde la computadora, por ese motivo se necesita un FTDI que nos permita esta capacidad.

- Tipo: Conversor USB-Serial UART
- Chip: FT232RL
- Comunicación: Transceptor RX/TX TTL COM Virtual RS232
- Voltaje: 3,3 o 5Vdc mediante jumper
- Pines: DTR, RX, TX, VCC, CTS, GND +All pin
- Protecciones: Fusible electrónico hasta 500mA
- Tamaño: 33 x 17 mm Pitch 2,54mm

iv. Servomotor: Micro Servo SG90^{iv}

Para el mecanismo de barrera, se eligió el SG90 debido a que era una opción más económica en comparación a otros servomotores y la fuerza entregada por este es suficiente para nuestro caso.

- Velocidad: 0.10 sec/60° @ 4.8V
- Torque: 1.8 Kg-cm @ 4.8V
- Voltaje de funcionamiento: 3.0-7.2V
- Temperatura de funcionamiento: -30 °C ~ 60 °C

v. Display: OLED 0.96" 128x64 ^v

Para el display se nos ocurrieron dos posibilidades, una es el display OLED 0.96'' que ya tenemos en nuestra disposición, otra opción es utilizar un LCD 16x2 con un módulo i2C integrado así solo necesitamos dos cables para la comunicación.

- Tamaño: 0.96"
- Color píxel: Blanco
- Ángulo de visión: más de 160 grados
- Plataformas compatibles: Arduino, 51 series, MSP430 series, STIM32/2, SCR chips
- Bajo consumo de energía: 0,04W durante el funcionamiento normal
- Soporte de voltaje: 3,3V - 5V DC

vi. Tira LED: WS2812B 5050^{vi}

Utilizar tiras LED RGB en comparación a LEDs tradicionales facilita el cambio automático de ruta en el caso que se llegue a implementar esa funcionalidad, además consideramos que se podría permitir al usuario elegir el color de la guía.

Las WS2812B son fácilmente programables y nos permite referenciar cada led individualmente.

- RGB LEDs individualmente accessible (30, 60, or 144 LEDs per meter)
- Control de color de 24-bit 16.8 millón de colores por píxel
- Interface digital de un cable
- 5 V voltaje de operación

XI. Presupuesto

	Cantidad	Precio Unidad (Pesos uruguayos)
Módulo sensor de proximidad / obstáculos MK0434	4	110
Microcontrolador ESP32-CAM	1	1110
Módulo adaptador de FT232RL	1	330
Micro Servo analógico TS90A 9g	4	330
LCD 16x2 1602 3V3 I2C y SP	1	520
Tira LED: WS2812B 5050	1	700
Módulo NodeMcu V3 ESP8266 WiFi	2	410
Total		5240

XII. Referencias

ⁱ<https://www.eneka.com.uy/robotica/sensores/optico/m%C3%B3dulo-sensor-de-distancias-hc-sr04-7391-7538-detail.html>

ⁱⁱ<https://www.hwlibre.com/esp32-cam/>

ⁱⁱⁱ<https://www.eneka.com.uy/robotica/modulos-comunicacion/m%C3%B3dulo-conversor-usb-serie-ttl-ft232rl-detail.html>

^{iv}<https://www.eneka.com.uy/robotica/motores/servomotores/motor-servo-micro-sg90-detail.html>

^v<https://www.eneka.com.uy/componentes/optoelectronica/displays/oled-0.96-128x64-7-pines-i2c-spi-blanco-detail.html>

^{vi}<https://cdn-shop.adafruit.com/datasheets/WS2812B.pdf>