

Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ciencias y Sistemas
Arquitectura de Computadores y Ensambladores 2
Catedrático: Ing. Gabriel Díaz
Auxiliar: Carlos Canté



Proyecto 1

Vehículo de entregas inteligente

Objetivos:

- Comprender los fundamentos de internet de las cosas mediante la implementación de una solución práctica.
- Diseñar un dispositivo que solucione una necesidad de los tiempos actuales.
- Diseñar un dispositivo que tenga conexión con aplicaciones externas mediante internet.
- Aprender a desarrollar una solución mediante la correcta implementación del framework de iot.
- Aplicar el uso de motores en el desarrollo de una solución IoT

Descripción

Debido a que el desarrollo del buzón inteligente fue un éxito y ayudó a reducir el contacto de los clientes con el repartidor de paquetes lo cual es muy importante en los tiempos actuales se busca dar un paso más en la automatización del proceso de recibimiento de paquetes entregados en las casa de los clientes, para esto se propone lo siguiente:

Se busca implementar un medio de transporte que sea capaz de llevar los paquetes depositados en el buzón inteligente hasta un destino como lo puede ser la puerta de la casa o la puerta de la habitación del cliente de forma automática, dicho transporte deberá de contar con una serie de funciones inteligentes que se detallaran mas adelante, ademas de contar con comunicación a una aplicación móvil por medio de internet.

Funciones:

El vehículo de transporte deberá de contar con las siguientes funciones:

1. Transporte del paquete desde el buzón hasta el punto de entrega:

- a. **Función:** El vehículo deberá de ser capaz de llevar el paquete recibido dentro del buzón hasta un punto de entrega de forma automática.
- b. **Implementación:** Esta función se llevará a cabo mediante el seguimiento de una línea trazada en el piso la cual le indicará al vehículo la ruta a seguir.

2. Retorno al buzón de entregas:

- a. **Función:** El vehículo deberá de ser capaz de regresar al buzón de entregas para poder recibir otro paquete al momento de que el actual fue entregado, esto debería poder realizarlo de forma automática sin intervención del usuario
- b. **Implementación:** Se utilizará la misma línea ya trazada para el recorrido solo que esta vez el vehículo deberá de ir en reversa, esto para evitar que el usuario deba de dar vuelta al vehículo para ir de regreso.

3. Medición de peso del objeto transportado:

- a. **Función:** El vehículo deberá de poder medir el peso del objeto que está transportando.
- b. **Implementación:** Se implementará un sensor de peso en el área donde el vehículo transportará la carga para poder realizar esta función.

4. Detección de obstáculos:

- a. **Función:** El vehículo deberá de ser capaz de detectar obstáculos en el recorrido tanto hacia el punto de entrega como de regreso al buzón de entregas, al momento de detectar uno el vehículo deberá de detener su marcha y continuar solo en el momento de detectar que el obstáculo ha sido retirado.
- b. **Implementación:** La detección de los obstáculos se implementa mediante el sensor ultrasónico **HC-SR04**.

5. Notificación de paquete en camino y de paquete entregado:

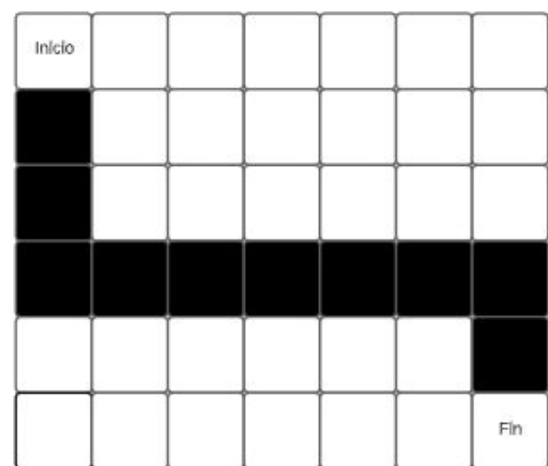
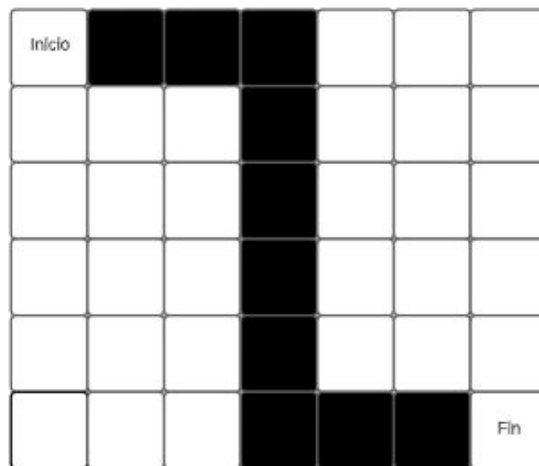
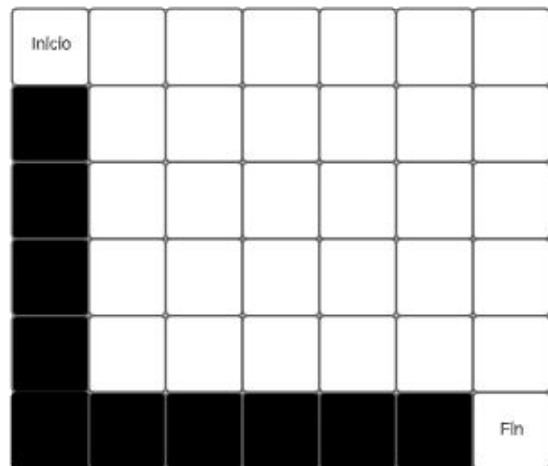
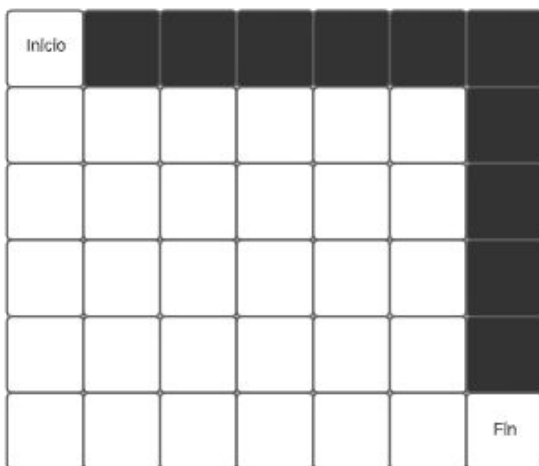
- a. **Función:** El vehículo deberá de ser capaz de notificar cuando salió del buzón con un paquete y cuando este ya fue entregado, así mismo el peso del mismo y la hora en la que se llevó a cabo la salida del buzón y la entrega.
- b. **Implementación:** Se deberá de mandar el dato al servidor para cuando el paquete sale del buzón y cuando este fue entregado para que el servidor la mande a la aplicación móvil y esta pueda notificar al usuario.

Restricciones del recorrido:

Se deberá de cumplir como mínimo los siguientes requisitos para el recorrido que hará el vehículo hacia el punto de entrega:

- El recorrido completo deberá de ser por lo menos 3 metros de largo
- Deberá de contener por lo menos 1 giro en L.

Ejemplos de recorrido válidos:



Conectividad:

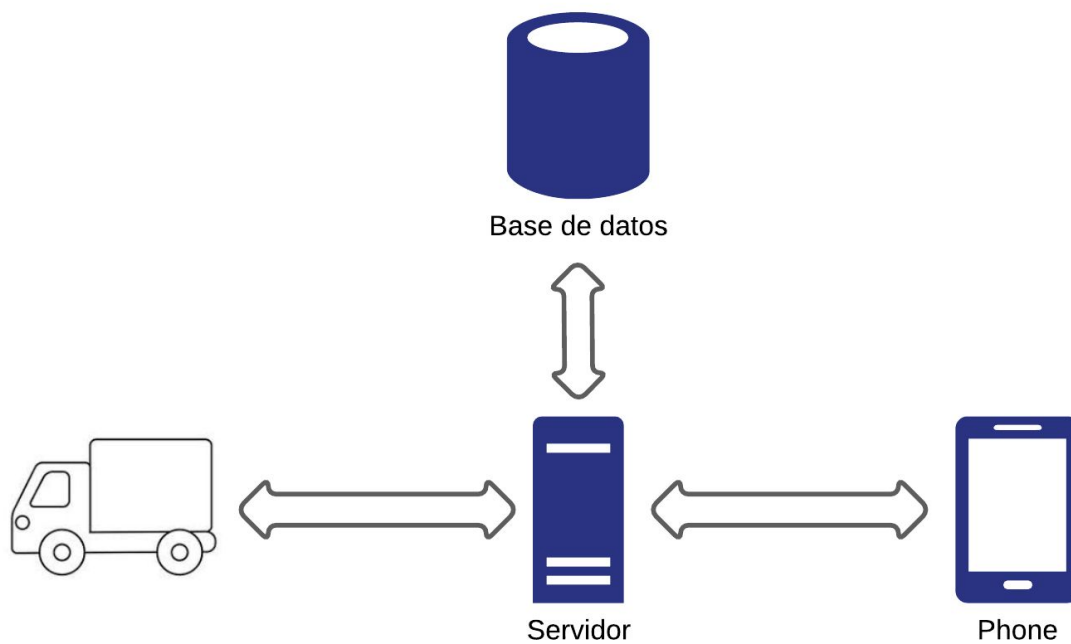
El vehículo deberá de poder conectarse a una aplicación móvil la cual servirá para visualizar los datos obtenidos que se detallan más adelante, para dicha comunicación se recomienda realizarla mediante API REST y un módulo WIFI para Arduino.

Los datos que este deberá enviar el vehículo son los siguientes:

- Ubicación actual en la que se encuentra el vehículo el cual puede ser:
 - Punto de partida (Dentro del buzón)
 - En el recorrido (Algún punto del recorrido ya sea de ida o retorno)
 - Punto de entrega.
- Estado actual del vehículo el cual puede ser:
 - En reposo.
 - En camino hacia el punto de entrega.
 - De regreso al al buzón.
 - Detenido por un obstáculo.
- Datos al momento de salir del buzón al punto de entrega los cuales son los siguientes:
 - Peso del objeto.
 - Hora de salida hacia el punto de entrega.
- Datos del paquete entregado los cuales se envían al momento de entregar el paquete:
 - Peso del objeto.
 - Hora de entrega del paquete.
 - Conteo de obstáculos encontrados.
- Datos del vehículo al momento de regresar al buzón:
 - Hora de regreso al buzón.
 - Conteo de obstáculos encontrados.

Diagrama de conectividad

A continuación se muestra un diagrama de la conectividad:



NOTA: Todos los datos recolectados serán utilizados en la práctica 2 por lo que es importante que su estructura y almacenamiento estén bien definidos además de que deben de estar disponibles después de la entrega del proyecto.

Aplicación móvil:

Se contará con una aplicación móvil la cual contará con una **única pantalla con un dashboard** en donde se mostrarán los siguientes indicadores:

- Ubicación del vehículo.
- Estado del vehículo.
- Cantidad de paquetes entregados.
- Total de obstáculos detectados.
- Peso promedio de los paquetes entregados.
- Tiempo promedio que ha tardado en llegar el vehículo desde el buzón hasta el punto de entrega.
- Tiempo promedio que ha tardado en llegar el vehículo desde el punto de entrega de regreso hasta el buzón.

Notificaciones de la aplicación móvil:

La aplicación deberá tener notificaciones tipo push para alertar al usuario de los siguientes casos:

- Notificación del vehículo al salir del buzón con los datos correspondientes:
 - Peso del objeto
 - Hora y fecha de salida hacia el punto de entrega.
- Notificación del vehículo al momento de llegar al punto de entrega:
 - Peso del objeto
 - Hora y fecha de llegada al punto de entrega
 - Conteo de obstáculos detectados.
- Notificación del vehículo al llegar de regreso al buzón:
 - Hora y fecha de llegada al buzón.
 - Conteo de obstáculos detectados.

Adicional a lo anterior el vehículo se deberá de poder colocar en cualquiera de los siguientes modos mediante la aplicación móvil:

- **Activo:** El vehículo funciona de forma normal.
- **Inactivo:** El vehículo permanece en reposo sin importar si llega un paquete, esto hasta el momento que se cambie a modo **Activo**.

Repositorio de GitHub:

Para el proyecto se seguirá utilizando el mismo repositorio que se utilizó en la práctica 1, el cual llevaba por nombre **ACE2_2S20_G#GRUPO**, ejemplo **ACE2_2S20_G12**, solo solicita lo siguiente:

- **Nombre del repositorio:** Deberá agregar al usuario **CarlosCante** como colaborador si no lo ha hecho.
- Hacer por lo menos 1 commit por semana durante el desarrollo.
- Todo código o documento que no se encuentre en el repositorio no será tomado en cuenta para la calificación.
- Se deberá de colocar todo lo correspondiente al proyecto en una carpeta llamada **Proyecto 1** dentro del repositorio.

Contenido obligatorio del repositorio:

- Código de Arduino utilizado.
- Código de la aplicación móvil
- Todo el código utilizado para la implementación del servidor como lo son las API
- Fotos del prototipo final
- Documentación completa.

Restricciones generales:

- La práctica se deberá realizar en los grupos formados para la práctica 1.
- Para el lenguaje a utilizar para el desarrollo de la aplicación móvil se recomienda ionic, flutter y react.
- Se deberá implementar un servidor para la conexión entre la aplicación y el buzón, la forma de implementación de este queda a discreción del estudiante.
- Se deberá tener una base de datos en la cual se almacenen los datos recolectados, el tipo de esta queda a discreción del estudiante.
- Se deberán respetar los roles definidos para cada estudiante durante el desarrollo.

Documentacion:

En la documentacion debera de llevar todo lo correspondiente al desarrollo tomando como base el Framework de iot, dicho lo anterior se solicita:

- Introduccion
- Bocetos del prototipo
- Pantallas de la aplicación móvil
- Capas del framework de iot.
- Link de un video del dispositivo funcionando.

La documentación deberá de ser presentada con el formato IEEE.

NOTA: Todo punto solicitado anteriormente deberá de ir dentro de la documentación si no no se tomará en cuenta.

Consideraciones:

- Se calificará solamente lo que sea completamente funcional.
- La comunicación entre el buzón y la aplicación deberá de estar implementada y funcional.
- Se deberán de mandar todos los entregables en la fecha establecida, de no ser así se tendrá una penalización del 50%.
- La documentacion se debera entregar en el formato IEEE, de no ser así se tendrá una penalización del 10%
- **Fecha de entrega: 20 de septiembre de 2020 antes de las 23:59**
- El link del repositorio deberá de ser entregado en la plataforma de UEDi en el área destinada para ello.
- También se deberá mandar tambien el link del repositorio al correo 2430897080101@ingenieria.usac.edu.gt con el asunto **[ACE2]Proyecto1_G#GRUPO** como medio de respaldo en caso la plataforma de UEDi tenga algún problema.