



“Reporte Proyecto EXPOCET”

ALUMNO:

Gonzalez Hernández Adrián Omar

PROFESOR:

EDUARDO CORNEJO VELAZQUEZ

SEMESTRE: 6 GRUPO: 3

Proyecto D-BI

Autores:

- Joselo Granillo Vargas
- Antuan Hernandez Acosta
- Alan Hasan Silva Herrera
- Evany Ruben Gonzales Rodriguez
- Angel Gabriel

Problema abordado:

El problema abordado en el proyecto es la necesidad de optimizar el servicio de dispensación de bebidas, reduciendo los tiempos de atención y mejorando la experiencia del usuario en entornos como exposiciones, eventos o locales comerciales. Tradicionalmente, los dispensadores de bebidas requieren intervención manual, lo que puede generar demoras, inconsistencia en las mezclas y dependencia de infraestructura física como botones o monedas.

Propuesta de solución:

Para resolverlo, el equipo creó un dispensador automático de bebidas controlado por WiFi. La propuesta es una caja con cuatro dispensadores que se maneja desde un microcontrolador ESP32. Lo más interesante es que el ESP32 crea su propia red y una página web a la que te puedes conectar con tu celular o computadora sin necesidad de descargar una app. Desde ahí, puedes servarte una bebida predefinida como un "Azulito" o una "Margarita" o crear tu propia mezcla decidiendo la cantidad exacta en mililitros de cada líquido. Para lograr esta precisión, el equipo calibró el sistema basándose en que cada dispensador entrega aproximadamente 10 mililitros por segundo, permitiendo al microcontrolador calcular exactamente cuánto tiempo debe activar cada bomba para obtener la cantidad deseada. Para que todo funcione de manera ordenada y confiable, incluso diseñaron y fabricaron su propia placa de circuitos PCB.

Aplicación con Autómatas y compiladores

El proyecto D-BI, aunque funcional con su implementación actual, podría beneficiarse significativamente de los conceptos de autómatas y compiladores para crear un sistema más robusto, flexible y profesional.

1. Implementación con Autómatas Finitos

Se podrían modelar los estados del dispensador usando Autómatas Finitos Deterministas (AFD):

Ejemplo:

Autómata Finito para el Sistema de Dispensación D-BI

Estados (Q):

- q_0 = Reposo (Estado inicial)
- Validando
- Dispensando
- Completado (Estado final)
- Error

Alfabeto (Σ):

- r = receta_válida
- i = receta_inválida
- d = dispensación_exitosa
- f = fallo_dispensación
- t = tiempo_agotado

Función de Transición (f):

f(Reposo, r) = Validando

f(Reposo, i) = Error

f(Validando, d) = Dispensando

f(Validando, f) = Error

f(Dispensando, d) = Completado

f(Dispensando, f) = Error

f(Dispensando, t) = Error

f(Error, r) = Validando

f(Completado, r) = Validando

Estado Inicial:

- q_0 = Reposo

Estados Finales (F):

- F = {Completado}

2. Diseño de un Lenguaje Formal para Recetas

Para permitir que los usuarios definan recetas complejas de manera estructurada, se podría crear un lenguaje formal. Este lenguaje tendría una sintaxis formal que permitiría expresar recetas de manera clara y sin ambigüedades.

Ejemplo con BNF:

```
{receta} ::= "RECETA" {nombre} ":" {instrucciones}
{instrucciones} ::= {instruccion} | {instruccion} {instrucciones}
{instruccion} ::= {líquido} {cantidad} "ml"
{líquido} ::= "LIQUIDO1" | "LIQUIDO2" | "LIQUIDO3" | "LIQUIDO4"
{cantidad} ::= {dígito} | {dígito} {cantidad}
{dígito} ::= "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
```

3. Análisis Léxico y Sintáctico

El proceso de interpretación de una receta seguiría las fases típicas de un compilador:

Análisis Léxico (Scanner):

Un analizador léxico se encargaría de reconocer los tokens del lenguaje:

- Palabras reservadas: RECETA, LIQUIDO1, LIQUIDO2.
- Números: Secuencias de dígitos para las cantidades
- Unidades: ml
- Símbolos: :, =

Ejemplo:

Entrada: "RECETA Margarita: LIQUIDO2 72ml LIQUIDO3 108ml"

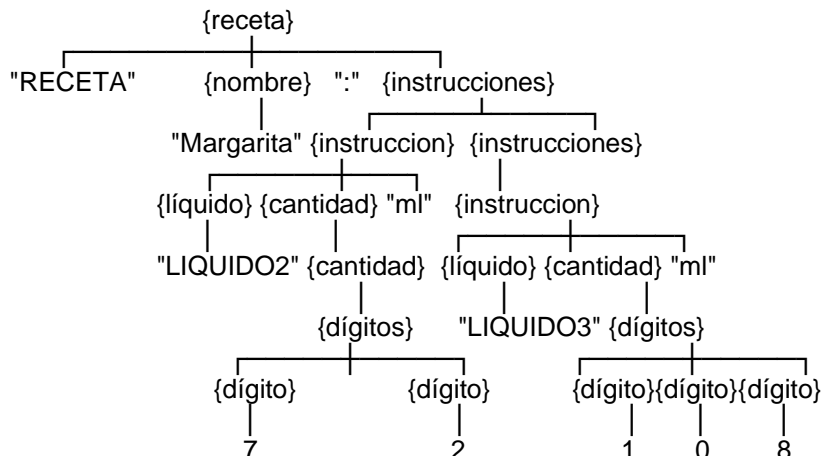
Tokens: [RECETA] [ID:"Margarita"] [:] [LIQUIDO2] [NUM:72] [ml] [LIQUIDO3] [NUM:108] [ml]

Análisis Sintáctico (Parser):

Utilizando la gramática definida, el parser construiría un árbol sintáctico que represente la estructura de la receta.

Verificaría que la secuencia de tokens sigue las reglas gramaticales.

Ejemplo:



Evidencia fotográfica:

