UNIVERSIDAD CATOLICA ANDRES BELLO

FACULTAD DE INGENIERIA

ESCUELA DE INGENIERIA INFORMATICA

CATEDRA DE ARQUITECTURA DEL COMPUTADOR

Proyecto

**Máquina de Estados Finitos**

**(FSM)**

Sección: 26414

Victoria Paciello

Miguel De Olim

Leonardo Ruíz

**Máquina de Estados Finitos**

Se puede entender con la explicación de que una máquina de estados finitos, es un modelo de comportamiento de un sistema, teniendo finitas posibilidades de estados, los cuales cambiarán bajo el cumplimiento de alguna condición, que se le pondrá el nombre de condición de transición. Cada uno de los estados determina el tipo de acción que la máquina lleva a cabo. Una transición indica el cambio de estado.

Todo sistema posee una representación gráfica, la representación utilizada para una máquina de estados finitos es un diagrama de estados, donde se observan por círculos los estados, con su identificación en el medio y unas flechas que indican la transición. Posee acción de ingreso, que es el nuevo estado que llega y se ejecuta una acción, análogamente posee una acción de salida que es el estado por el cual la máquina termina su ejecución.

A continuación, se presenta un diagrama del proyecto que posee 8 estados diferentes, los cuales son:

* Búsqueda (Primera acción después de la acción de entrada)
* Llevar al microondas
* Devolver
* Nueva búsqueda
* Recargar
* Ir a la base
* Aleatorio
* Muerte (Última acción antes de la acción de salida)

Como fue explicado con anterioridad, es importante poseer una transición para los cambios de estado, estos deben cumplir una condición por lo cual se explicará dependiendo del estado y observando el diagrama:

El primer estado siendo “Nueva búsqueda” posee la condición de si hay o no comida pendiente, en caso de ser Verdadera la condición, la máquina cambiará de estado a “Búsqueda”, siguiendo por este camino, se debe cumplir la condición de llegar al objeto para cambiar de estado a “Llevar al microondas” que en caso, continuando debe esperar a que la condición de que la comida esté caliente se cumpla para volver a cambiar de estado a “Devolver”, suponiendo que la comida no esté caliente, será la condición para que el estado cambie a “Muerte” y todo el sistema. Siguiendo el camino donde el estado actual es “Devolver” para cambiar de estado sólo se debe cumplir que llegue al destino y terminará en “Nueva búsqueda” otra vez.

En el caso en el que el estado actual es “Búsqueda” y la condición anterior no se cumpla es y se cumpla que la batería sea menor a 400 (Batería < 400), lo cual hace que la máquina cambie de estado a “Ir a la base” en el momento en el que se cumpla la condición de transición de llegar a la base, la máquina cambiará de estado a “Recargar” y cuando su condición de transición se cumpla, que es la batería recargada, vuelve a donde empezó que es el estado de “Búsqueda”.

Si en el estado actual “Ir a base” no se cumple la condición anteriormente descrita, entonces en algún momento se cumplirá que la batería sea igual a cero (Batería = 0) y será la condición para que la máquina cambie de estado a “Muerte”.

El último caso es en el cual el estado actual sea “Nueva búsqueda” y se cumpla la condición en que no haya comida pendiente, se cambiará al estado “Aleatorio” lo cual, este estado sólo cambiará si se cumple que la batería, nuevamente sea igual a cero (Batería = 0) y el estado cambie a “Muerte”.

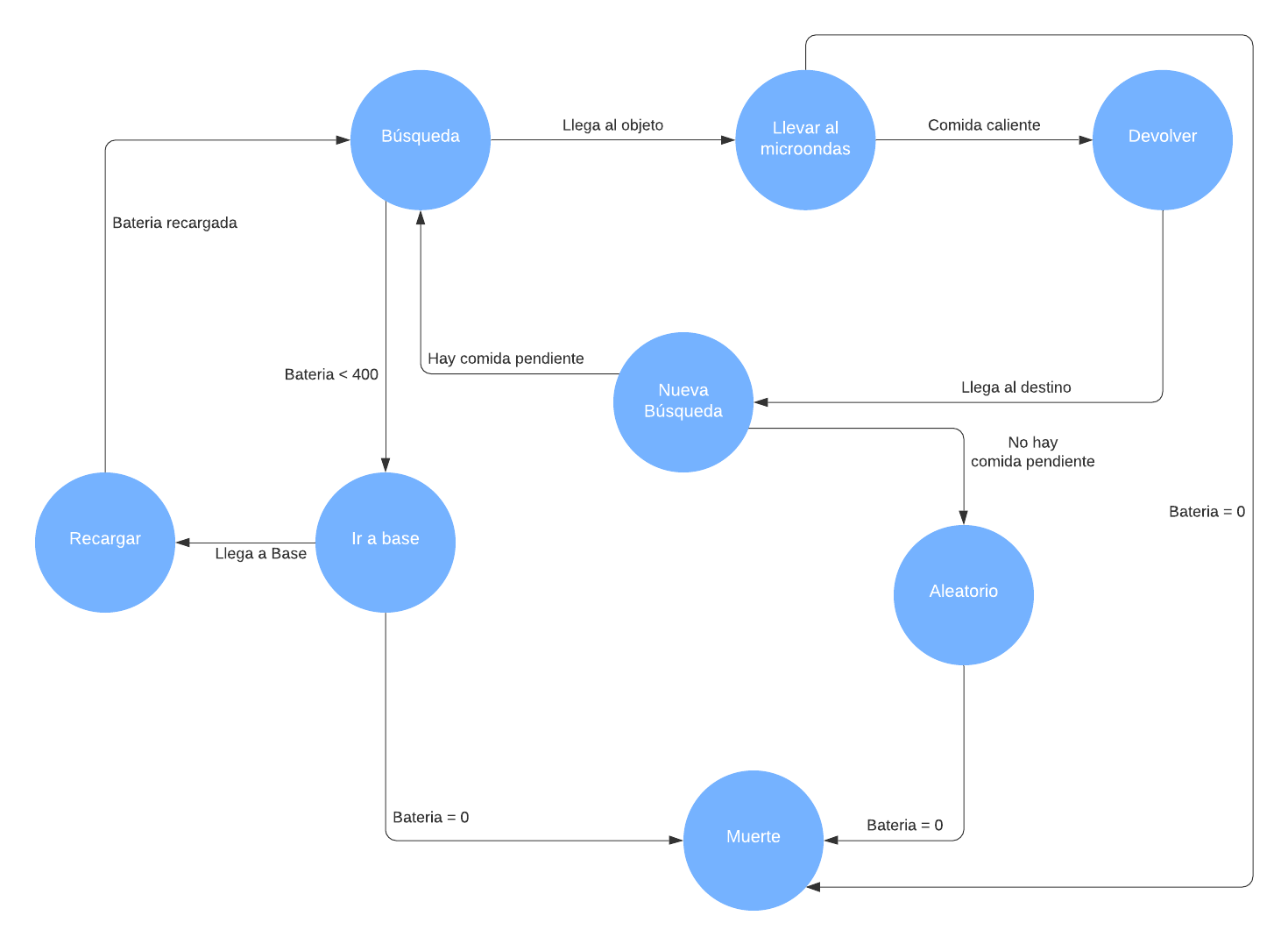
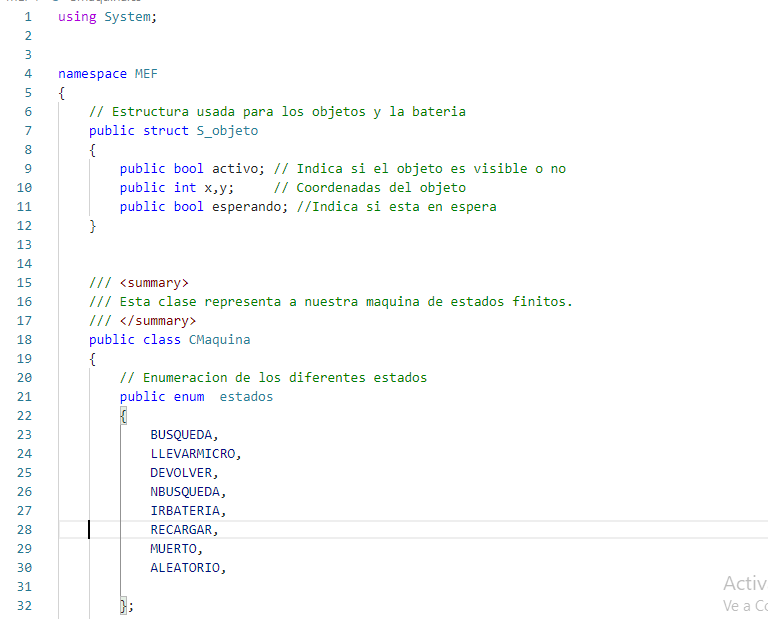


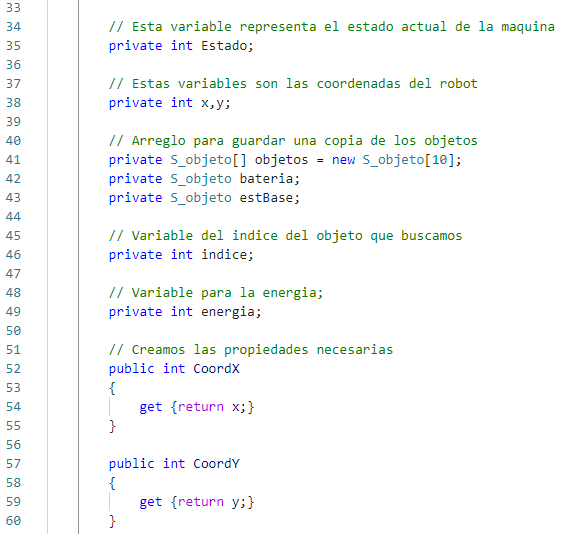
Fig. 1

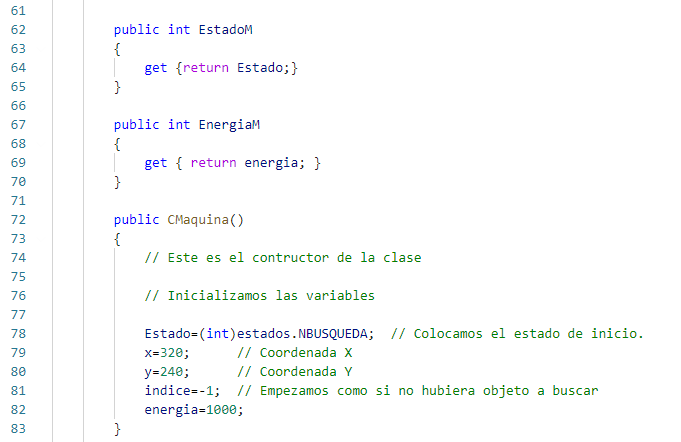
El proyecto está orientado a una simulación de una simulación para un protocolo para calentar la comida en la Universidad Católica Andrés Bello (UCAB) a través del uso de un dron, realizando el recorrido que se muestra en el diagrama, representado en la figura 1, que cada lugar nuevo en donde se obtiene la comida, son ciertas zonas de recarga, que estarán ubicadas a lo largo de la universidad. Los estudiantes colocarán la comida en el dron, este se dirige a calentar la comida en algún microondas y se regresa.

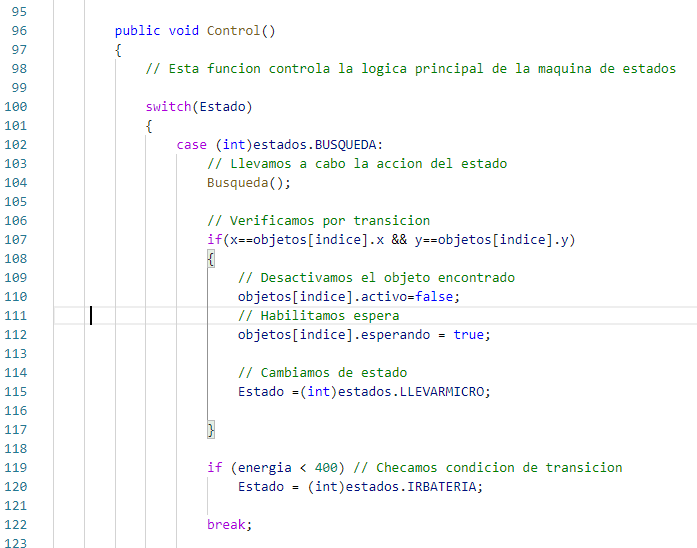
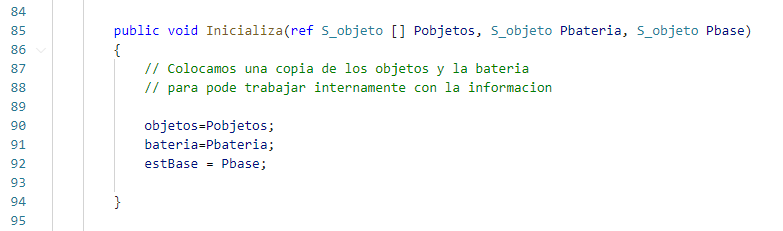
Esto con el fin de agilizar el movimiento y flujo de los estudiantes a la hora de la comida, para acortar el tiempo en el que los estudiantes hacen la fila para calentar la comida y desplazarse con comodidad. A pesar de que, parece un juego es una simulación prototipo del proyecto que ciertos estudiantes emprendedores les encantaría implementar.

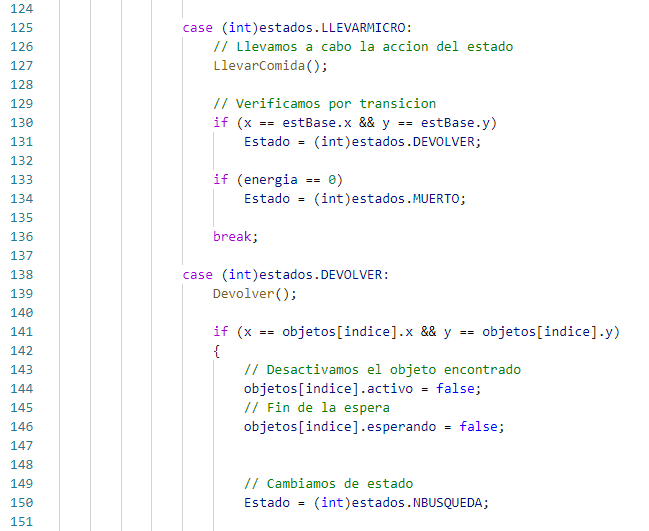
A continuación, se explica paso a paso el código del sistema:



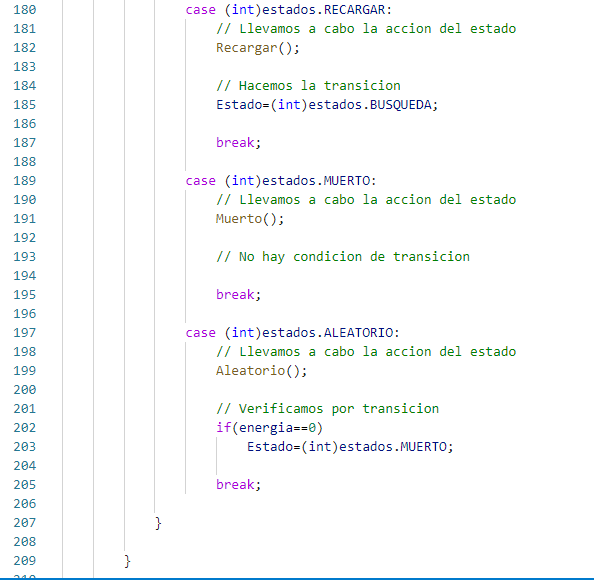


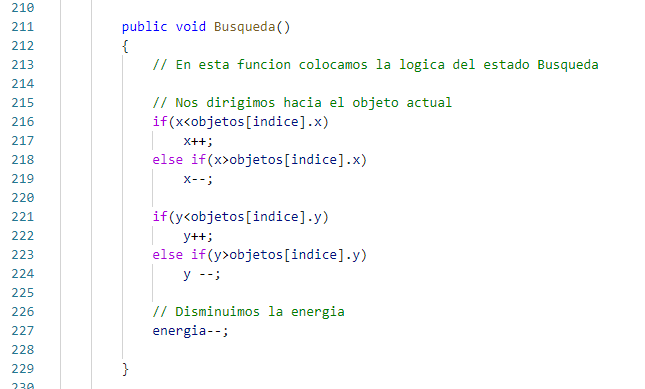


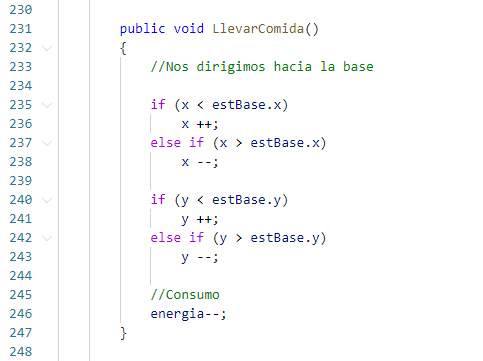


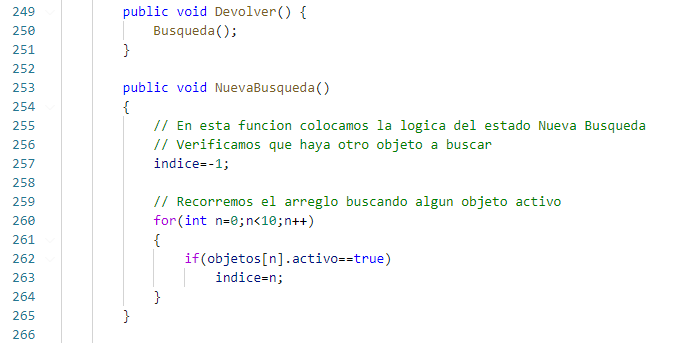


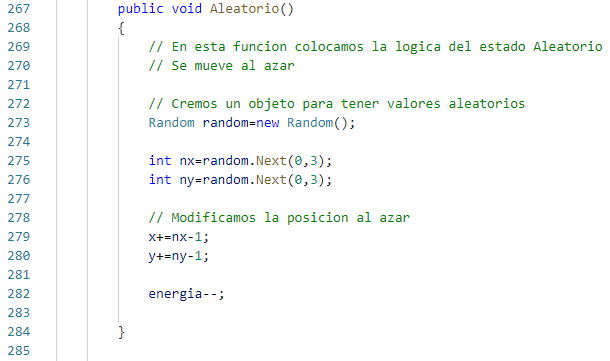


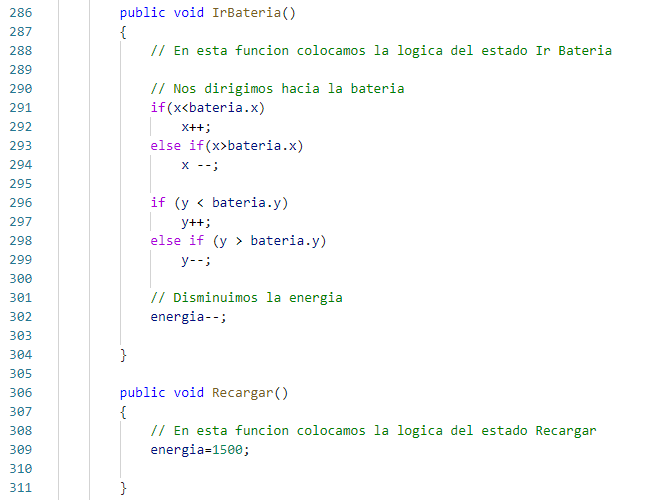


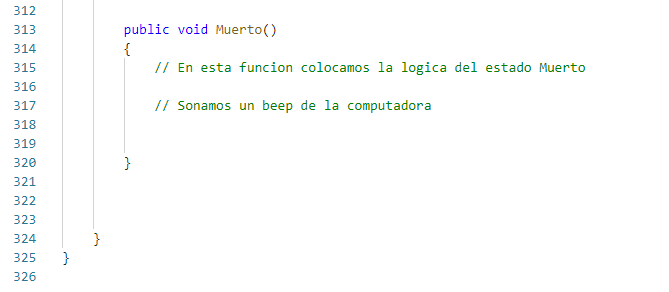








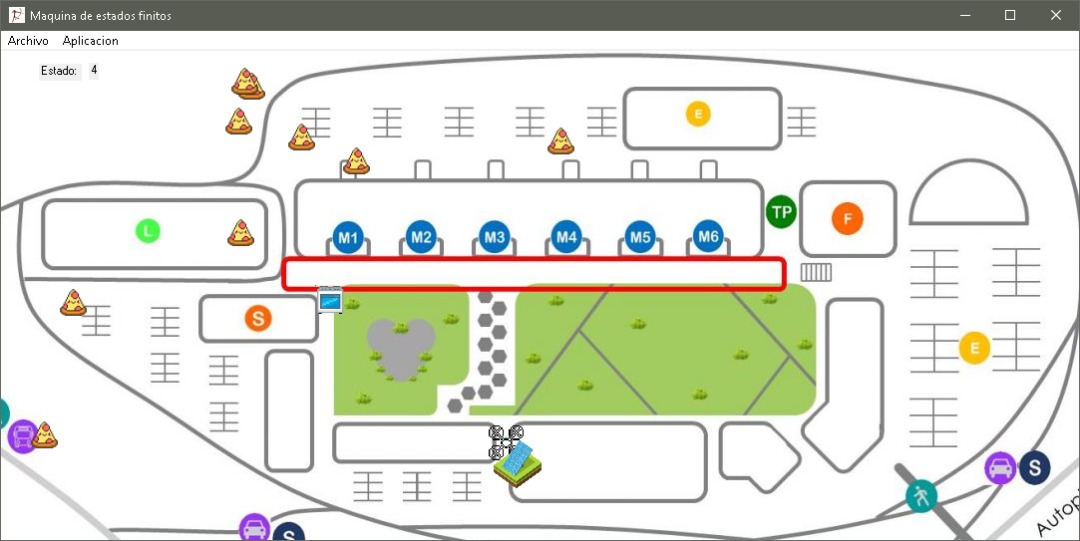


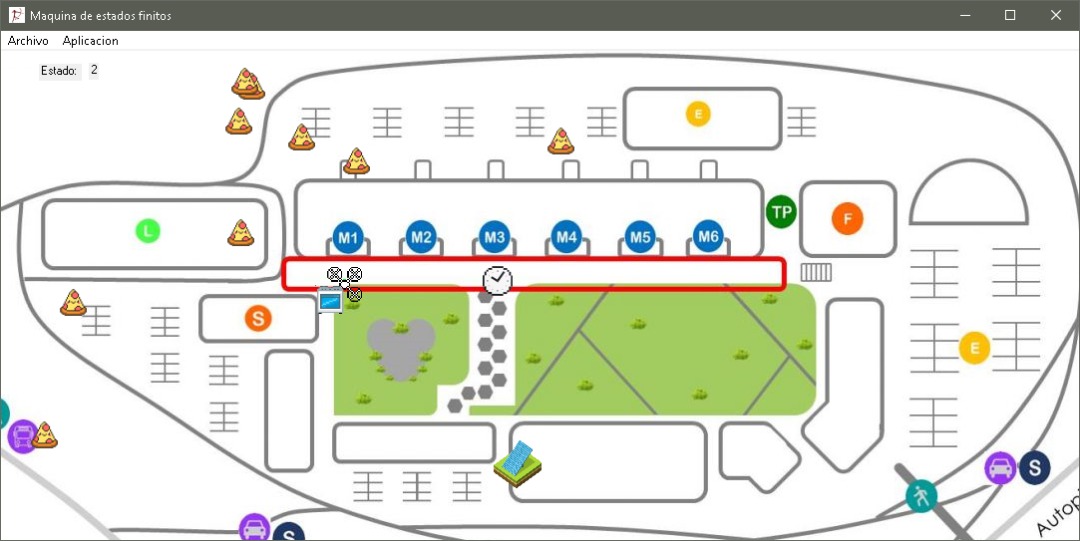
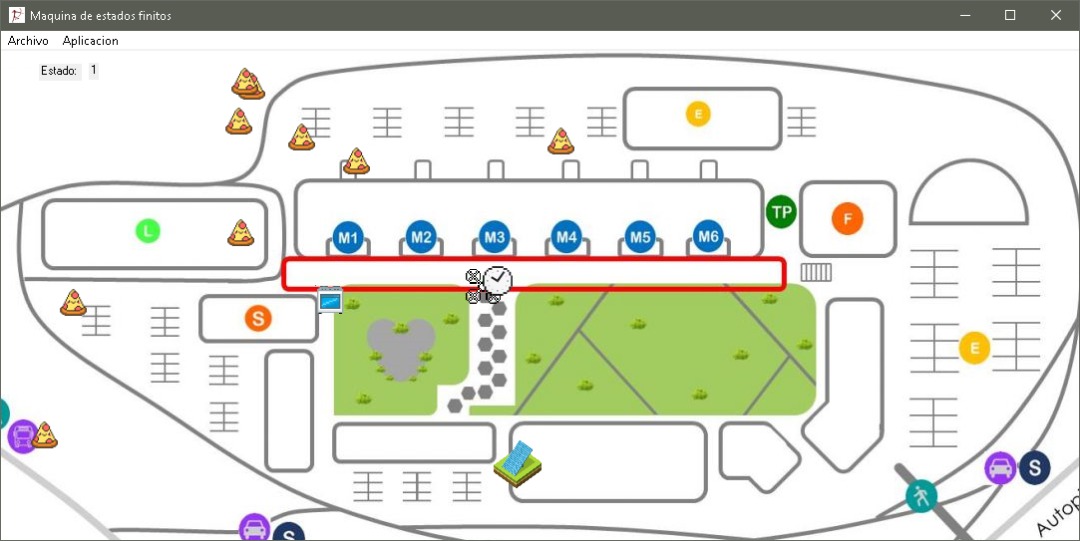
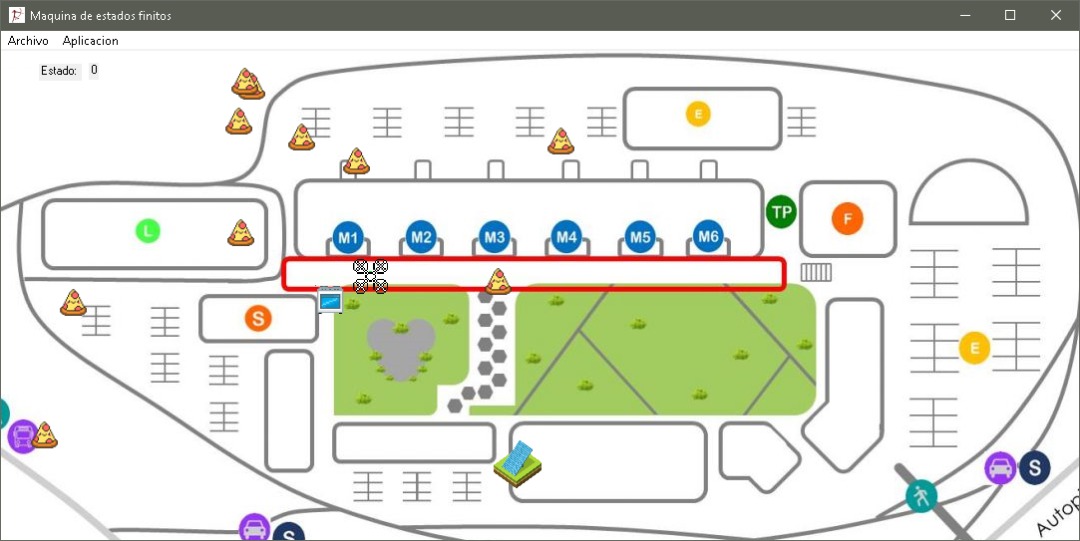
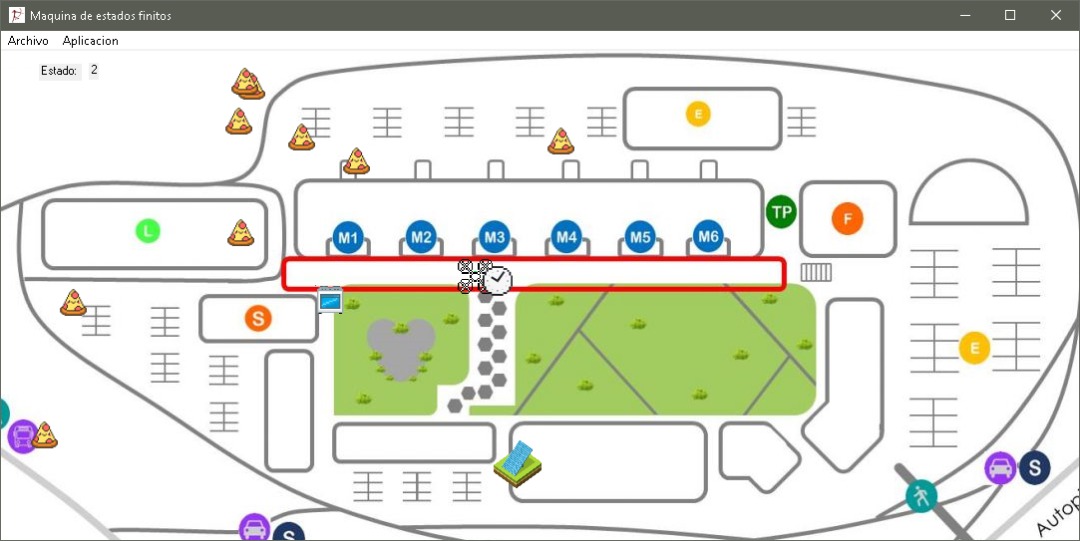
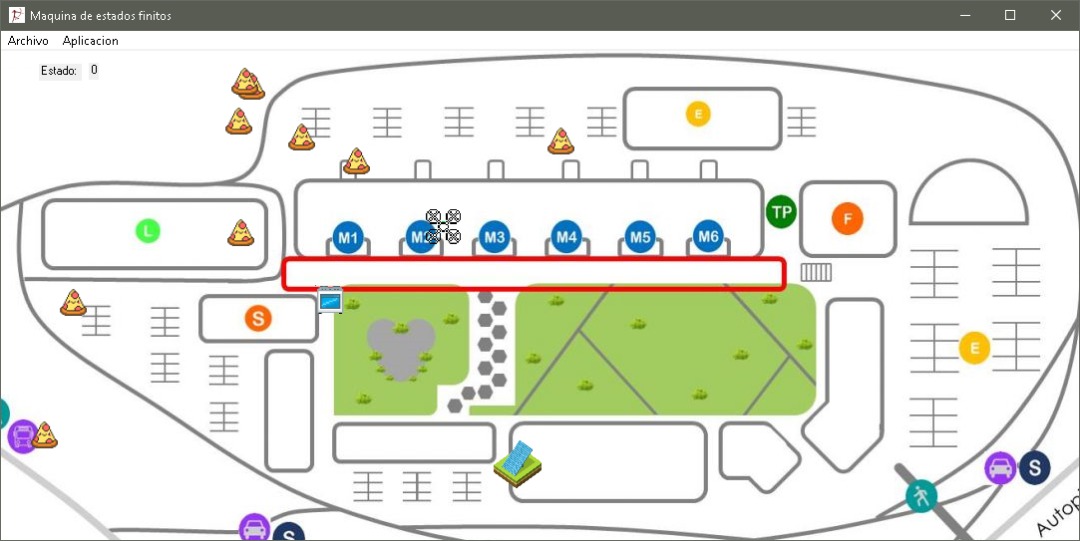
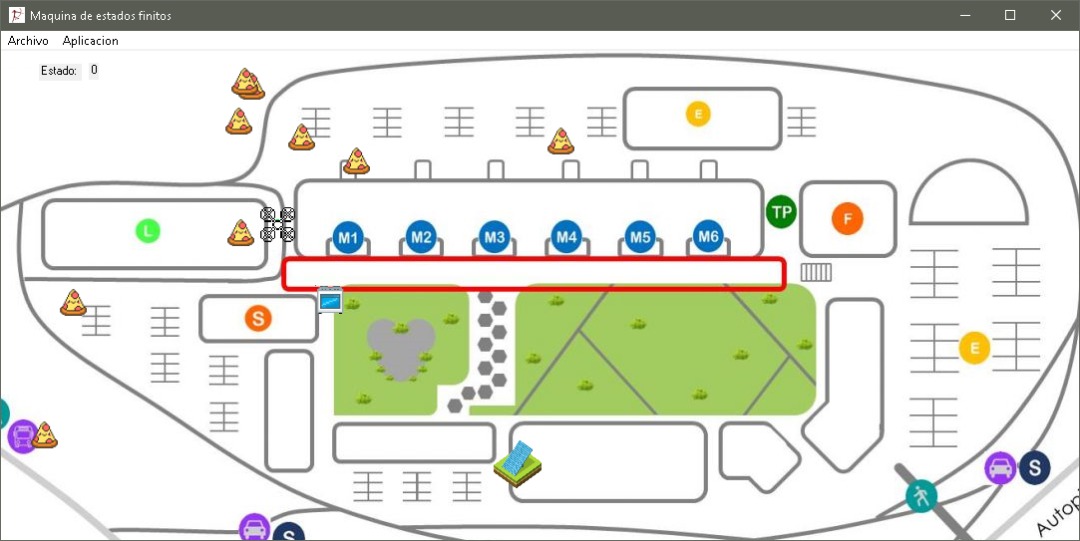
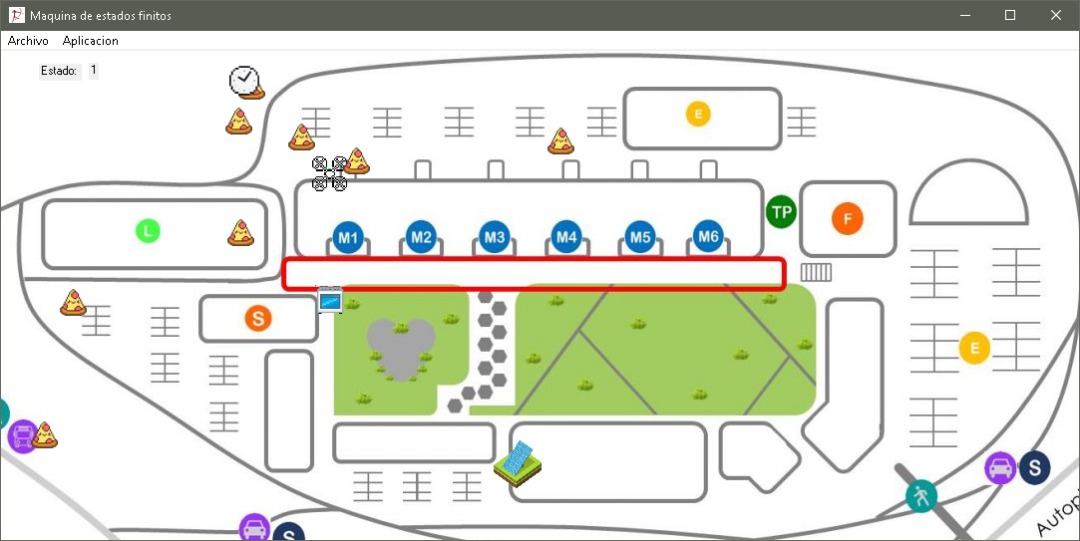
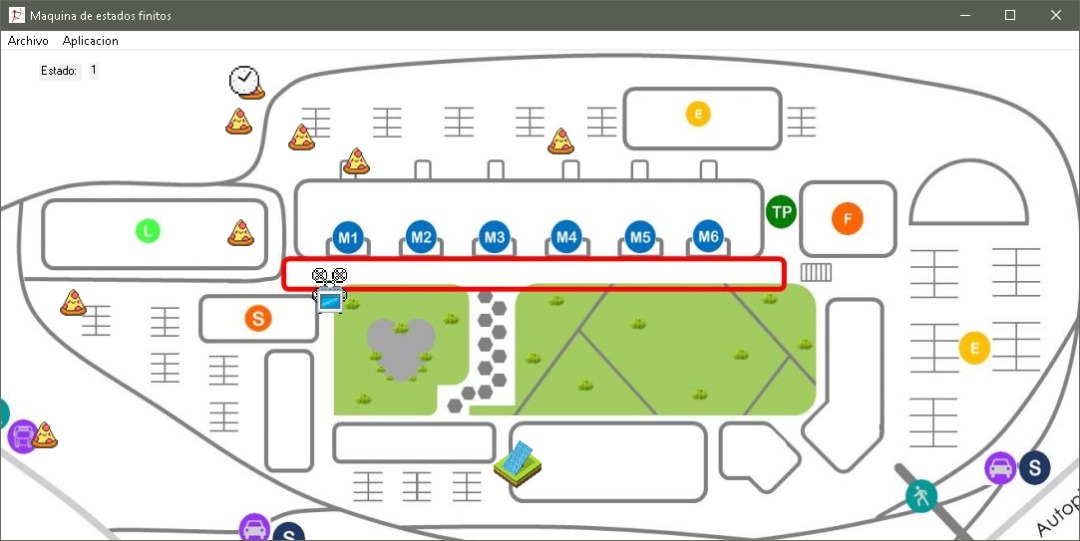


En cada parte del código está, con comentarios (antecedidos por //) la explicación de cada procedimiento y/o función utilizada, la creación de los estados y las condiciones de transición.

A continuación, una recopilación del funcionamiento visual del código anterior agregado con una parte visual que oculta la abstracción de la máquina de estados finitos, realiza el trabajo demostrado en el diagrama de estados:

Se puede observar la posición inicial de la máquina de estado





Por último, se ve que el estado de muerte o el de salida se denota con un icono de fuego simulando una explosión.



Como se mencionó a lo largo del informe, el potencial que poseen las máquinas de estados finitos son ilimitadas, el desarrollo de cualquier sistema puede ser basado en el comportamiento lógico de estas, por lo cual dan pie a que todo se realice con cada vez menos abstracción.

Las simulaciones pueden ser modeladas de diferentes formas, a través de modelos matemáticos, diagramas de estado, entre otros, porque estos son abstracciones de la pseudorealidad que se percibe y se intenta que sea más acercada a la realidad en sí, por lo que posee muchas variantes, a lo cual a veces no se está conscientes de todas. Es importante aclarar que las representaciones de con máquinas de estados finitos, son representaciones bastante abstractas y dependiendo del modelo pueden ser más o menos complejas.