

# **Advice Machine**

**Tercer Proyecto Programado de Taller de  
Programación**

**I semestre 2020**

**Estudiantes:**

**Anthony Chaves Achoy  
Fernando Flores Moya**

**Profesor:**

**Jeff Schmidt Peralta**

**Año: 2020**

## **Tabla de Contenidos**

<b>Portada.....</b>	<b>1</b>
<b>Tabla de Contenidos.....</b>	<b>2</b>
<b>Introducción.....</b>	<b>3</b>
<b>Descripción del Problema.....</b>	<b>4</b>
<b>Modelo de Objetos.....</b>	<b>5</b>
<b>Análisis de Resultados.....</b>	<b>6</b>
<b>Reglas del Grupo.....</b>	<b>10</b>
<b>Dificultades Encontradas.....</b>	<b>11</b>
<b>Bitácora.....</b>	<b>12</b>
<b>Estadística de tiempos.....</b>	<b>15</b>
<b>Conclusiones.....</b>	<b>16</b>

# Introducción

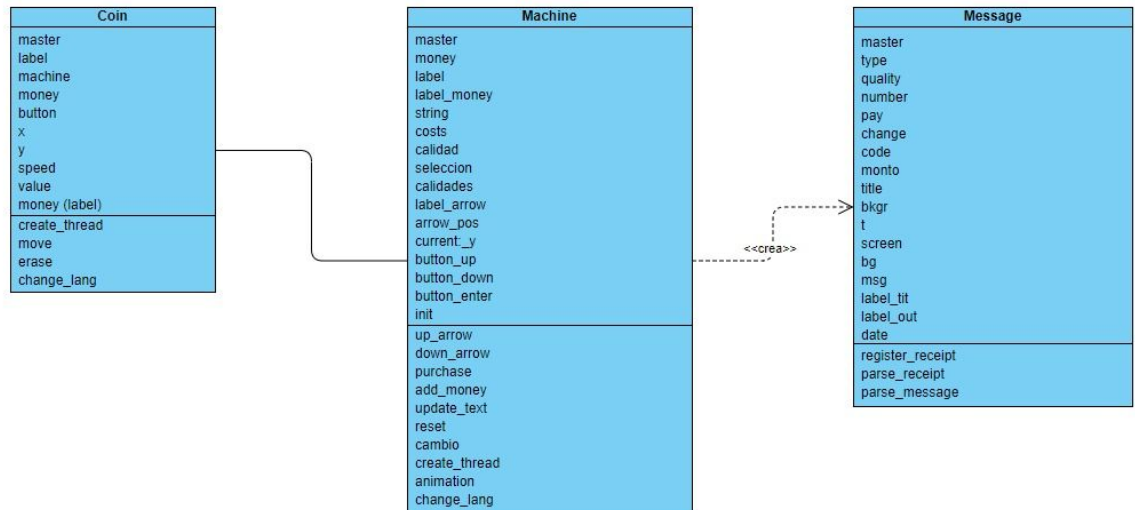
El área de las máquinas dispensadores ha avanzado considerablemente conforme la electrónica y la programación lo han hecho, pasando de ser únicamente sistemas de mecanismos en sus inicios, a incorporar sistemas embebidos con una complejidad mucho mayor, permitiendo un aumento de funciones bastante considerable por medio de funcionalidades como el guardado no volátil, la incorporación de programas de un nivel más alto, Internet of Things y componentes electrónicos que interactúa con todos estos aspectos. Partiendo de lo anterior, una *Advice Machine* hace uso del guardado no volátil y de un sistema de selección simple para que un usuario pueda pagar por un consejo, o en su defecto un chiste o un dicho, y posteriormente recibirlo impreso. El procesamiento de las entradas proporcionadas por el usuario y la salida se logra gracias a una tarjeta de desarrollo como un Arduino o una Raspberry Pi, donde el software juega un papel fundamental.

## Descripción del Problema

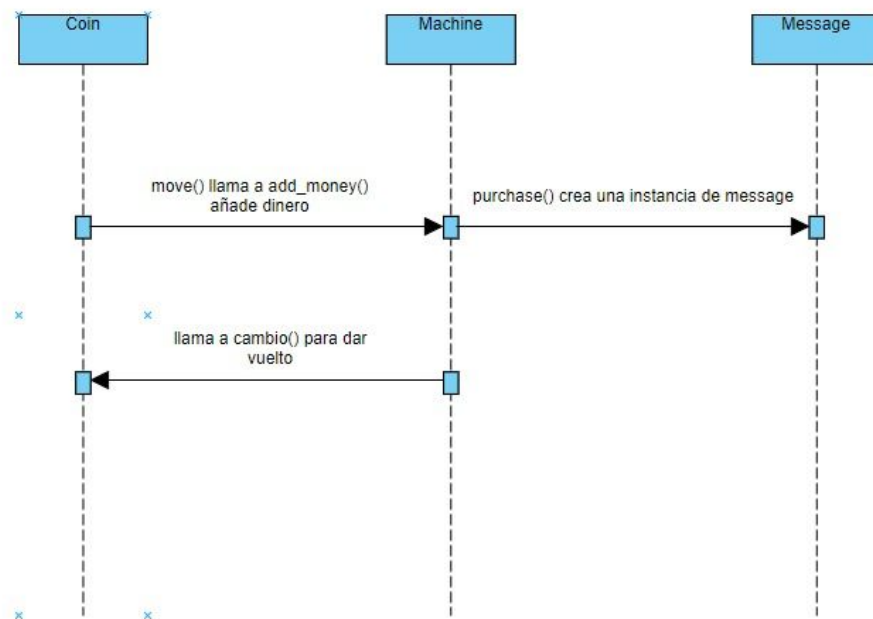
Basado en la explicación de la sección anterior sobre lo que es una *Advice Machine*, el problema que se presenta en este proyecto es simular uno de estos dispositivos haciendo uso de herramientas de interfaz gráfica proporcionadas por el lenguaje de programación Python, donde el hardware y la interacción del usuario con objetos reales, en este caso dinero, son reemplazados por botones, animaciones, pop ups y otros elementos de la interfaz gráfica. Por otro lado, existen varias funcionalidades relacionadas al guardado en archivos de texto y el procesamiento de esta información para conocer la eficacia de las ventas de ciertas frases, los montos generados, las transacciones, reinicio de ventas y demás aspectos. Existe también una “segunda interfaz”, con las opciones de administrador, para un mejor control de la máquina. Parte de la lógica debe poder manejarse con un modelo de clases, detallado más adelante.

# Modelo de Objetos

## Diagrama de Clases

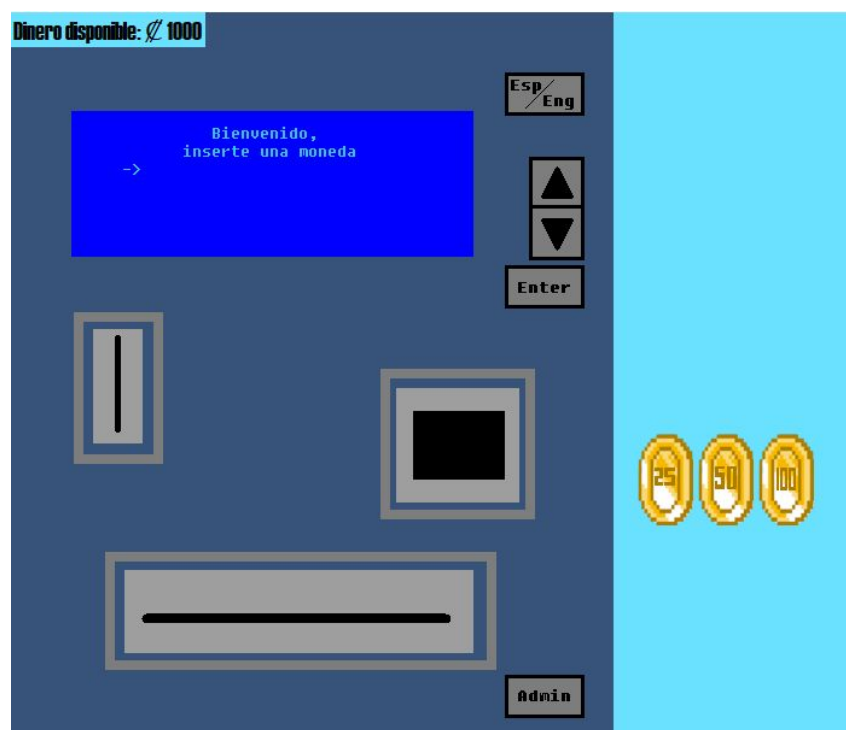


## Diagrama de secuencia



## Análisis de Resultados

Partiendo con la interfaz gráfica inicial, podemos observar la máquina y una sección que contiene las monedas y el dinero restante (Figura 1). Esta división busca generar en la simulación un efecto de división entre la interacción con la máquina en sí y la interacción con el dinero, que el usuario insertaría a la máquina con sus propias manos, lo cual se logró con una animación que traslada las monedas al aceptador de monedas (Figura 2).

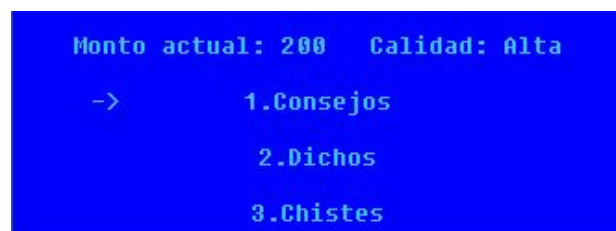


*Figura 1: Interfaz de usuario*

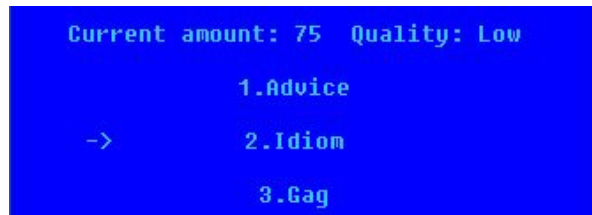


*Figura 2: Animación de la moneda desplazándose*

Otro aspecto a destacar es la forma en la que el usuario utiliza la máquina, ya que al insertar la moneda el display pasa de un “estado de espera”, a un estado de selección, donde con tres botones, arriba, abajo y *enter*, se selecciona el tipo de mensaje y se muestra constantemente su calidad según el dinero invertido, además de condicionar el idioma de las salidas según la elección del usuario (Figuras 3 y 4). Al presionar *enter* la máquina realiza una animación de impresión y genera otra ventana que muestra el mensaje (Figura 5). Si existe un sobrante de dinero, este se devuelve al usuario por medio de una moneda.



*Figura 3: Display en estado de selección en español*



*Figura 4: Display en estado de selección en español*



*Figura 5: Ventana que muestra el mensaje*

Por otra parte, en los archivos de texto se optó por la utilización de un formato similar a “dato1.dato2.dato3...”, especificando que es lo que cada dato representa en la parte superior de la pantalla (Figura 6). Esta representación permite simplificar muchísimo el procesamiento y separación de los datos con la lógica del programa, ya que de forma similar al proyecto anterior, una estructura similar a una máquina de estados utiliza los puntos y saltos de línea para guiarse y conocer qué es lo que representa cada dato obtenido. Todos los parseos del programa giran en torno a este concepto, incluyendo los datos que se muestran, o bien, se borran por medio de la sección de administrador (Figuras 7 y 8).

```
Formato
Tipo.Código.Mensaje.Precio.Ventas
1.1.Afronta tus miedos.50.0
1.2.Nunca pierdas la esperanza.50.0
1.3.Haz ejercicio cada día.50.0
1.4.Menos es mas.50.0
```

*Figura 6: Formato del archivo de texto de mensajes*



```
def generate_report():
    f = open("mensajes.txt", "r")
    f2 = open("messages.txt", "r")
    words = f.read()
    jump = 0
    str_out = ""
    params = []
    mat_out = []
    for x in range(2):
        for i in range(len(words)):
            if jump >= 2 and words[i] != "\n":
                if words[i] == ".":
                    params += [str_out]
                    str_out = ""
                elif words[i] != ".":
                    if words[i] != "/":
                        str_out += words[i]
                    else:
                        str_out += " "
            elif words[i] == "\n":
                if jump >= 2 and int(str_out) > 0:
                    params += [str_out]
                    mat_out += [params]
                    params = []
                    str_out = ""
                jump += 1
        params = []
        jump = 0
    words = f2.read()
```

Figura 7: Código para obtener las ventas por mensaje

Type	Code	Message	Messages sold	Amount in sales
1	9	Busca un nuevo hobby	1	€100
1	11	Cada cosa a su tiempo y en su sitio	1	€100
2	15	En boca del mentiroso, lo cierto es dudoso	1	€200
1	12	Your future needs you, your past does not	1	€100
2	8	All is fair in love and war	1	€100

Figura 8: Tabla de mensajes obtenida por medio del parseo anterior

Para ver de una mejor forma la funcionalidad del proyecto puede acceder al link <https://youtu.be/A62kIAxnFnw>, donde encontrará una demostración de todo lo anterior.

Por otra parte, el proyecto de GitHub se encuentra en el link [https://github.com/FernandoF87/Proyecto\\_Programado\\_3](https://github.com/FernandoF87/Proyecto_Programado_3).

## Reglas del Grupo

1. Avisar constantemente sobre los avances, cambios y los push a github. También implica explicar a la otra persona el código añadido.
2. Ayudar con las pruebas del código del otro miembro para darle feedback sobre errores.
3. Procurar realizar el trabajo de forma asincrónica con respecto a los horarios del otro, ya que esto nos facilita bastante todo lo relacionado al *merge* del código, ya que deja de ser necesario la mayor parte del tiempo.
4. Aceptar y aplicar los consejos acerca de eficiencia, buenas prácticas de programación.
5. Mantener una división de trabajos equitativa y basada en las aptitudes de cada miembro.
6. Respetar la división de trabajos acordada e intentar mantener una buena estética en la interfaz

## Roles

**Fernando:** Partiendo de que en este proyecto el trabajo fue un poco más mixto en términos de implementación de la interfaz y la lógica, Fernando se hizo cargo principalmente de las animaciones, la lógica relacionada a la forma en que el usuario interactúa con la máquina por medio de una simulación con monedas, la sección de administración y una parte del parseo, principalmente en lo relacionado a las facturas y el corte de ventas. Por otra parte, en la documentación Fernando realizó el video.

**Anthony:** Las tareas de Anthony inician con el diseño de la máquina en sí y la forma en la que se simula el hardware de esta por medio de botones y un “display”, incluyendo la selección de idioma y la lógica estructurada en torno a esto. Además de esto, también realizó la parte de los parseos relacionada a los mensajes y la forma en la que se muestran al usuario.

## Dificultades encontradas

En general no tuvimos muchas dificultades durante este proyecto debido a la experiencia previa con manejo de archivos de texto, strings e interfaz gráfico, Sin embargo, un problema que surgió durante el desarrollo de la pantalla de administrador fue el generar el reporte de ventas de una manera limpia ya que inicialmente se mostraban todos los datos de cada uno de los mensajes sin importar si tuvieran ventas o no. Para solucionar este problema lo que se hizo fue crear un algoritmo que generara una lista compuesta por sublistas que contienen los datos de cada mensaje que tuviera mínimo una venta. De esta forma se facilitó el manejo de cada uno de los datos del mensaje ya que se podían manejar como elementos individuales de una lista. A su vez se tuvieron algunos problemas con el espaciado de los datos en el reporte de ventas pero lo que se hizo fue establecer una longitud específica para los mensajes, en este caso 94 caracteres, y cada mensaje escribiría con exactamente esa cantidad de caracteres agregando espacios en los casos que el mensaje fuera más corto (Figura 9).

```
if language:
    to_write = "Tipo\tCodigo\tMensaje\t\t\t\t\t\t\t\t\t\t\tMensajes vendidos\tMonto ventas\n"
else:
    to_write = "Type\tCode\t\tMessage\t\t\t\t\t\t\t\t\t\t\tMessages sold\tAmount in sales\n"
for i in mat_out:
    for j in range(len(i)):
        if j != 4:
            if j == 2:
                while len(i[j]) < 94:
                    i[j] += " "
            if j != 3:
                to_write += i[j] + "\t"
            else:
                to_write += i[4] + "\t\t"
        else:
            to_write += "€" + str(int(i[3])*int(i[4]))+"\t"
to write += "\n"
```

Figura 9: Escritura del reporte de ventas con el debido espaciado.

# Bitácora

## Fernando

Fernando 4 de agosto de 2020- 4:30pm a 6:30pm (Programación): Creé la clase Coin y le añadí animaciones básicas a la moneda para que simule la moneda ingresando en la máquina

Fernando 5 de agosto de 2020- 4:30pm a 6:30pm (Programación y Diseño): Creé la clase Machine, modifiqué animaciones para coincidan con la máquina y cambié los labels que muestran el dinero actual de la máquina y del usuario.

Fernando 6 de agosto de 2020- 3:30pm a 5:30pm (Programación y Diseño): Creé la clase Message que define los mensajes y su comportamiento, también agregué funcionalidad a la clase Machine para que genere una animación simulando la impresión del mensaje y manejo de hilos para la creación de un objeto Message.

Fernando 8 de agosto de 2020- 11:00am a 12:30pm (Diseño): Agregué cambios en la pantalla de administrador para que se viera más limpio y ordenado. También agregué funcionalidad al botón de apagar dispensador y dejé la base para los otros dos botones. Finalmente creé la función de mensajes pop-up para errores o notificaciones al usuario.

Fernando 11 de agosto de 2020- 3:00pm a 6:00pm (Programación y Pruebas): Creé las funciones de parseo de facturas y generación de reportes de ventas. Agregué funcionalidad a los dos botones restantes en la pantalla de administrador (corte de ventas y reporte de ventas) y realicé pruebas para comprobar la eficacia de los parseos y las funciones.

Fernando 12 de agosto de 2020- 3:00pm a 3:30pm (Diseño y pruebas): Corregí errores visuales de los mensajes y mejoré la ubicación y estética de las frases.

Fernando 12 de agosto de 2020- 5:00pm a 6:00pm (Documentación Externa): Realicé los diagramas de clase y de secuencia.

Fernando 13 de agosto de 2020- 10:00am a 12:00pm (Diseño): Mejoré la estética general de los reportes de ventas.

Fernando 13 de agosto de 2020- 4:00pm a 6:30pm (Documentación Externa): Trabajé con Anthony en la documentación externa, también grabé y edité el video que muestra los resultados y funcionalidades del programa.

Fernando 14 de agosto de 2020- 2:00pm a 4:00pm (Documentación Interna): Trabajé en documentación interna.

## **Anthony**

Anthony 4 de agosto de 2020- 9:00pm a 11:00pm (Diseño): Hice los primeros diseños e ideas para el manejo de la interfaz gráfica de la máquina, considerando la interacción con el usuario, las monedas y el hardware de la máquina, simulados con elementos que provee Tkinter.

Anthony 5 de agosto de 2020- 7:00pm a 9:00pm (Diseño, programación): Agregué el display y diseñé la interfaz con simula los botones de la máquina y la forma en la que esta interactúa con el usuario.

Anthony 5 de agosto de 2020- 9:00pm a 11:00pm (Programación): Trabajé sobre la clase Machine de Fernando para implementar todas las funciones necesarias para poder mover y seleccionar un tipo de mensaje utilizando una flecha en el display.

Anthony 7 de agosto de 2020- 8:00pm a 10:00pm (Programación): Agregué las funciones de cambio de idioma y los cambios que estas producen, además de generar un reporte preliminar de ventas que usaremos más adelante.

Anthony 8 de agosto de 2020- 10:00pm a 12:00am (Investigación): Busqué las 54 frases necesarias para la implementación del primer parseo, el cual tomaría un mensaje aleatorio según un rango de 18 en 18, con 6 frases de cada calidad por cada categoría.

Anthony 10 de agosto de 2020- 11:00pm a 2:00am (Programación): Agregué el parseo de los mensajes y logré que se mostraran en una ventana nueva, sin embargo la estética debería mejorarse bastante.

Anthony 11 de agosto de 2020- 11:00pm a 12:00am (Diseño, Investigación): Mejoras de la estética de los mensajes por medio de fondos personalizados para cada categoría. Además cambié algunas fuentes y utilicé variables para modificar el color del texto. Por último, agregué los mensajes en inglés.

Anthony 12 de agosto de 2020- 12:00am a 2:00am y de 9:00pm a 12:00am (Programación, Diseño, Pruebas): Mejoras en la lógica del programa en todo relacionado a la toma de los parseos, la estructura de los datos en forma de matriz para el reporte. Además hice bastantes pruebas para ver si se toman los valores con ventas mayores a cero y de esta forma poder tabularlos con una mejor estética.

Anthony 13 de agosto de 2020- 4:00pm a 10:00pm (Documentación Externa): Realicé mi parte de la documentación externa incluyendo desde la Introducción hasta el Análisis de problemas.

## Estadística de tiempos

<b>Tipo de actividad</b>	<b>Tiempo total en horas (suma del de ambos miembros)</b>
<b>Investigación</b>	3.5
<b>Programación</b>	16.5
<b>Diseño</b>	11.5
<b>Pruebas</b>	2.5
<b>Documentación Interna</b>	2
<b>Documentación Externa</b>	9.5
<b>Total</b>	45.5

# Conclusiones

## **Por parte de Fernando**

Durante el proceso de desarrollo de este proyecto entendí la importancia y utilidad del uso de archivos de texto y como puede ser incorporado de manera relativamente simple y efectiva para el guardado de datos proporcionados por una interfaz gráfica. Previo a este proyecto no entendía muy bien cómo manejar los datos brindados por este tipo de archivos, pero logré aprender principalmente de mi compañero la flexibilidad de estos y las maneras en que se puede incorporar para crear una mejor experiencia para el usuario. También logré experimentar la importancia que tiene un buen modelado de objetos, definición de métodos y la interacción entre objetos debido a que en gran parte esto fue lo que nos permitió ejecutar las acciones de pagos, animaciones e impresiones que de lo contrario hubieran requerido instrucciones más complejas y específicas.

## **Por parte de Anthony**

Partiendo de que en esta ocasión el trabajo fue un poco más mixto en el sentido de lógica y estética, el eje del aprendizaje de este proyecto es la versatilidad sobre el comprendimiento de un problema en sus distintas dimensiones, en mi caso en el de diseño de interfaz. Este campo es bastante complejo ya que implica pensar bastante como un potencial usuario, donde lo entendible y simple de la interfaz a primera vista juega un papel fundamental. Cada widget debe estar estratégicamente colocado en la pantalla, siendo esto un proceso bastante difícil de idear por lo ingenioso que llega a ser en algunos casos. Aprendí que definitivamente el consejo de “menos es más” es una regla de oro en este tipo de trabajos.