



Pontificia Universidad
JAVERIANA
Cali

Introducción a la Programación

Proyecto de Semestre

Sebastian Izquierdo Saavedra

Isabela Gutiérrez Reyes

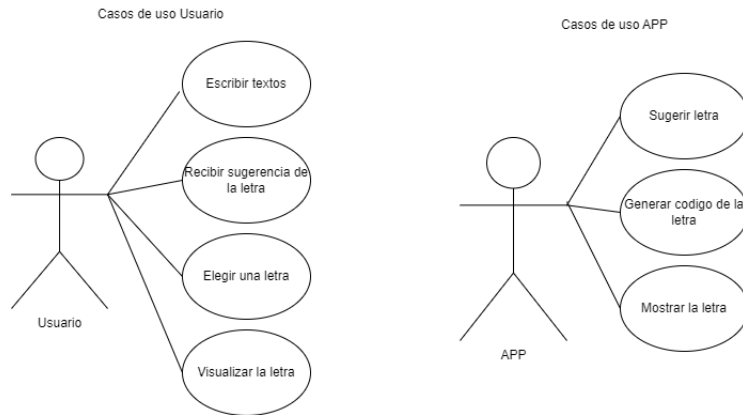
Joshua Mendez Ospina

Mario Julian Mora Cardona

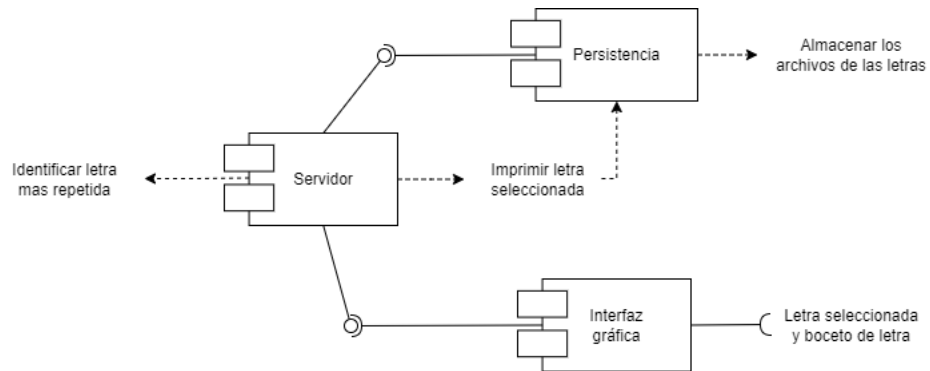
Mayo 18 2023

1 Diagramas UML

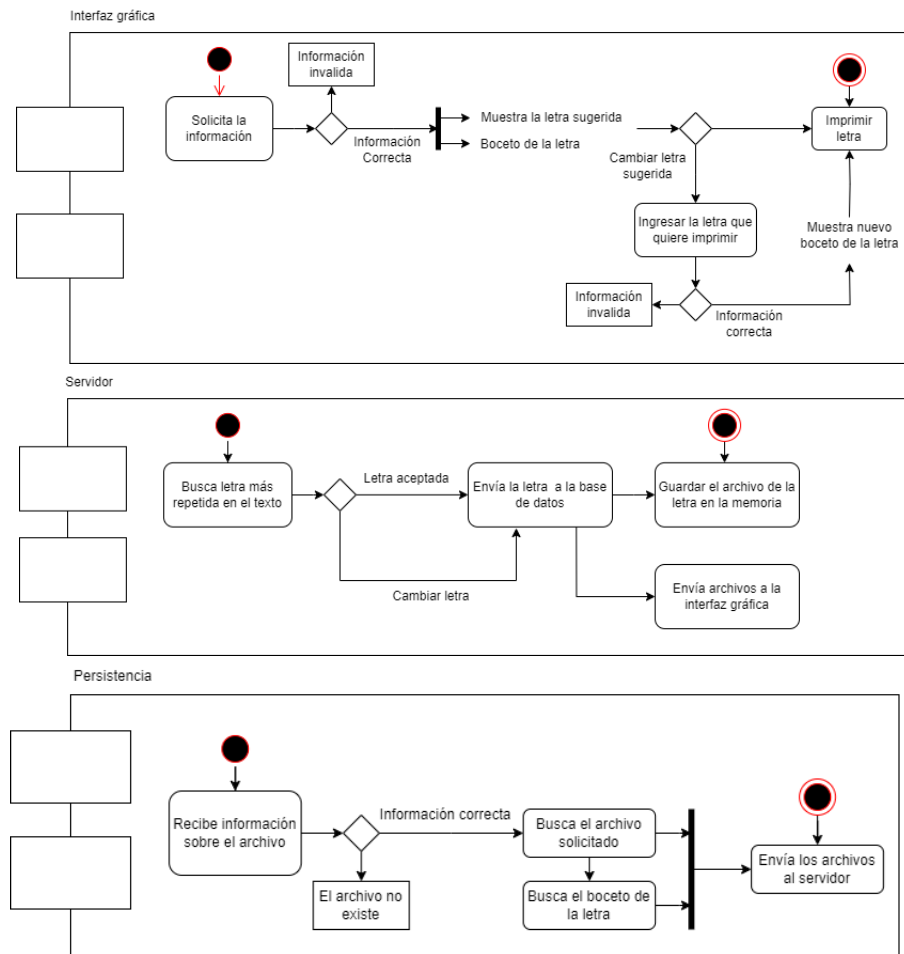
1.1 Diagrama de Casos de Uso



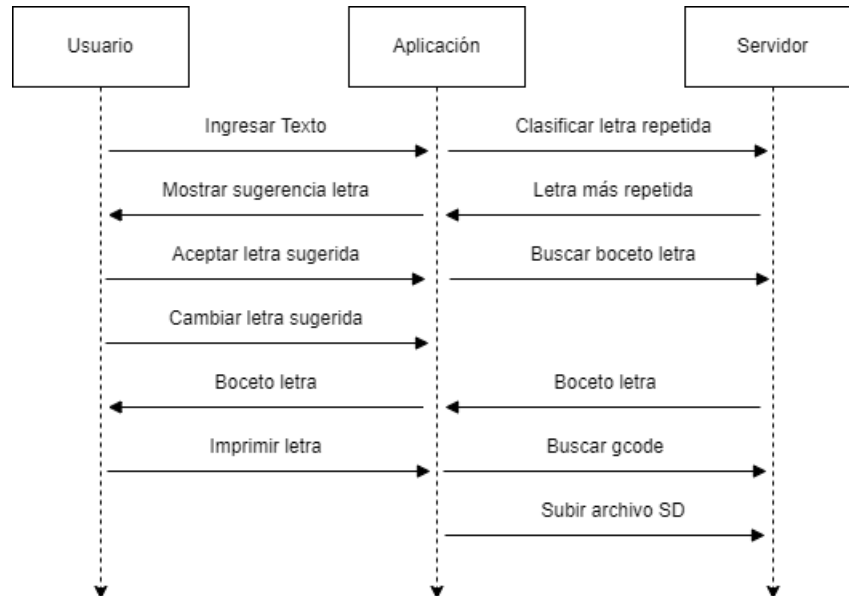
1.2 Diagrama de Componentes



1.3 Diagrama de Actividades



1.4 Diagrama de Secuencia



2 Contratos

2.1 Interfaz gráfica

```
ventana = ctk.CTk() # configuracion de la ventana
ventana._set_appearance_mode("dark") # cambia el modo de la apariencia
ventana.title("Garras, Perros y Llaveros S.A. - Recuerdos") # asigna titulo a la ventana
ventana.geometry("1270x720") # dimensiones de la ventana
letra_elegida = ""
```

- La función `funcionconfirmar()` muestra un cuadro de diálogo para seleccionar una carpeta y copia el archivo "letra.gcode" en la ruta seleccionada.
- La función `funcionconfirmar()` valida que letra ha sido elegida y con base a la letra elegida entrega el grafico de como se veria impresa en 3D y abre la opcion de confirmar impresión. usa la variable `letraelegido` de tipo string , usa un boton y un label elementos de la interfaz.
- La función `funcionaceptar()` recibe una nueva letra ingresada, la valida y actualiza la letra elegida. Utiliza las variables globales `nuevaLetra` y `letraelegida` y son de tipo string.
- Interfaz: `entradaletranueva` (campo de entrada de texto), `mensajeerror` (etiqueta para mostrar mensajes de error), `letraseleccionada` (etiqueta que muestra la letra seleccionada). Retorna False si la nueva letra no es válida.

```
def funcion_aceptar():
    global nuevaLetra, letra_elegida

    # Obtener la nueva letra ingresada
    nuevaLetra = entrada_letranueva.get()

    # Reiniciar el mensaje de error
    mensaje_error = ctk.CTkLabel(ventana, text="", font=("consolas", 18, "bold"))
    mensaje_error.place(x=500, y=550)
    mensaje_error.configure(text="")

    # Validar la nueva letra
    if nuevaLetra.isdigit():
        mensaje_error.configure(text="Eso no parece una letra!")
        return False
    if len(nuevaLetra) > 1 or len(nuevaLetra) == 0:
        mensaje_error.configure(text="Recuerda que las letras tienen 1 caracter!")
        print('es más de un caracter')
        return False
    else:
        print('correcto')
        letra_elegida = nuevaLetra.upper()
        mensaje_error.configure(text="Se cambió exitosamente")
        letra_seleccionada.configure(text="Letra seleccionada: " + str(letra_elegida))
```

- La función `cambiarletra()` muestra la interfaz para cambiar la letra seleccionada. Utiliza las variables globales `letraelegida`, `entradaletranueva`, `aceptar`, `letraseleccionada` y `cambiarletraboton`.
- Interfaz: `Ingreseletranueva` (etiqueta para indicar que se ingrese la nueva letra).

```
def cambiarletra():
    global letra_elegida, entrada_letranueva, aceptar, letra_seleccionada, cambiarletraboton

    # Interfaz cambiar letra
    Ingrese_letranueva = ctk.CTkLabel(ventana, text="Ingrese la letra: ", font=("consolas", 18, "bold"), width=400, height=10)
    Ingrese_letranueva.place(x=400, y=400)
    entrada_letranueva = ctk.CTkEntry(ventana, font=("consolas", 12, "bold"), width=100, height=50)
    entrada_letranueva.place(x=500, y=450)
    aceptar = ctk.CTkButton(ventana, text="Aceptar", font=("arial", 18), height=50, width=200, command=funcion_aceptar)
    aceptar.place(x=700, y=450)
    cambiarletraboton.place_forget()
```

- La función `algoritmosugerencia()` analiza el texto ingresado y sugiere la letra más repetida como recordatorio. Utiliza las variables globales `letraelegida`, `letraseleccionada`, `cambiarletraboton`, `imprimir` y `cantidadrepeticionL`.

- Interfaz: Etiqueta que muestra el resultado de la sugerencia.

```
# elementos de la ventana principal

# Título de la ventana
Titulo = ctk.CTkLabel(
    ventana, text="Garras, Perros y Llaveros S.A.", font=("consolas", 25, "bold"))
Titulo.place(x=340, y=30)

# Etiqueta para ingresar texto
Ingreso_texto = ctk.CTkLabel(ventana, text="Ingrese las palabras , frases o poemas de su preferencia: ", font=(
    "consolas", 18, "bold"), width=400, height=10)
Ingreso_texto.place(x=500, y=100)

# Campo de entrada de texto
entrada_texto = ctk.CTkEntry(ventana, font=(
    "consolas", 12, "bold"), width=550, height=50)
entrada_texto.place(x=500, y=150)

# Etiqueta para mostrar el resultado
resultado = ctk.CTkLabel(ventana, text="", font=("consolas", 14, "bold"))
# Botón para enviar el texto y ejecutar el algoritmo de sugerencia
Enviar = ctk.CTkButton(ventana, text="Enviar", font=(
    "arial", 18), command=algoritmo_sugerencia, height=50, width=200)
Enviar.place(x=500, y=230)
resultado.place(x=710, y=230)

# Iniciar la ventana principal
ventana.mainloop()
```

2.2 Código letras

```
ExteriorListaX = [100, 120, 120, 115, 115, 105, 105, 100, 100] # Lista que almacena las coordenadas X del contorno externo de las letras
ExteriorListaY = [150, 150, 110, 110, 130, 130, 110, 110, 150] # Lista que almacena las coordenadas Y del contorno externo de las letras
InteriorListaX = [] # Lista que almacena las coordenadas interiores X, todos estos valores se calculan en el código de abajo
InteriorListaY = [] # Lista que almacena las coordenadas interiores Y, todos estos valores se calculan en el código de abajo
rellenoHorizontalista = [] # Lista que almacena cómo iría el relleno horizontal de las letras
rellenoVerticalista = [] # Lista que almacena cómo iría el relleno vertical de las letras
```

- Especificaciones impresora
- Empezamos en 0s (Centro)
- Halla la diferencia que hay entre una coordenada y la que sigue para conseguir el valor por el cual se va a multiplicar. También imprime las coordenadas.

```

while ExteriorPosX < len(ExteriorListaX) and ExteriorPosY < len(ExteriorListaY):
    # Calcula la diferencia para que no se salga de la lista
    if DiferenciaExterior < (len(ExteriorListaX)) and DiferenciaExterior < (len(ExteriorListaY)) and (DiferenciaExterior != len(ExteriorListaX)):
        DiferenciaExterior = DiferenciaExterior + 1
        E1 = abs(int(ExteriorListaX[ExteriorPosX]) - int(ExteriorListaX[DiferenciaExterior]))
        E2 = abs(int(ExteriorListaY[ExteriorPosY]) - int(ExteriorListaY[DiferenciaExterior]))
        if E1 != 0 and E2 == 0:
            RE = E1 * 0.03
        if E2 != 0 and E1 == 0:
            RE = E2 * 0.03
        E = E + RE
        E = round(E, 3)

```

- Para el contorno interior se utilizan las coordenadas ya ingresadas y se le resta o suma 0.3
- Dependiendo de donde esté ubicada, si el valor menos 0.3 es menor a 100 entonces le suma 0.3
- Si la coordenada más 0.3 es mayor 120 entonces le resta 0.3 y se toma un punto medio
- Para el contorno en Y se toma como valor de referencia el 110 para que lo evalúe en 3 casos diferentes y se realiza el mismo proceso
- El resultado se almacena en las listas InteriorListasX e InteriorLista y se hace el mismo proceso para hallar los valores de E.
- Ya con los contornos se hace el relleno vertical y horizontal en donde se ingresa la coordenada inicial de X y este va sumando 0.3 hasta llegar a 120 que es la coordenada final y se establecen los valores de Y, tanto el punto inicial como el punto final y se imprime un G0 y un G1 intercalados para que siempre vuelve a la parte superior de la letra. Se ingresa el último valor que tomó

E y la distancia que va a recorrer y cada que avance 0.3 en X esta distancia se va a multiplicar por 0.03 para conseguir el valor final de Y.

- Se realiza el mismo procedimiento para el relleno horizontal solo que las coordenadas que varían van a ser las de Y. Se realiza todo este proceso hasta que el valor en Z sea 2 lo cual equivale a las 8 capas que contiene la letra y se obtiene el código G que se envía a la impresora.

3 Manual de Usuario

¡Bienvenido a la aplicación "Garras, Perros y Llaveros S.A."! Este manual te guiará a través de los pasos para utilizar la aplicación de manera eficiente y obtener los resultados deseados.

Requisitos del sistema

- Python instalado en tu sistema.
- Las bibliotecas necesarias: PIL (Python Imaging Library), tkinter, customtkinter, shutil y os.

Asegúrate de cumplir con estos requisitos antes de comenzar.

Ejecutar la aplicación

1. Abre una terminal o línea de comandos en tu sistema.
2. Navega hasta la ubicación donde tienes guardado el archivo de código.
3. Ejecuta el archivo de código utilizando el comando `python main.py`.

Interfaz principal:

Después de ejecutar la aplicación, se abrirá la ventana principal de "Garras, Perros y Llaveros S.A.". En esta ventana, encontrarás los siguientes elementos:

1. Título: En la parte superior de la ventana, verás el título "Garras, Perros y Llaveros S.A.".
2. Etiqueta de entrada de texto: Justo debajo del título, verás una etiqueta que te indica que ingresas las palabras, frases o poemas de tu preferencia.

3. Campo de entrada de texto: A continuación de la etiqueta, hay un campo donde puedes ingresar tu texto.
4. Botón "Enviar": Al lado del campo de entrada de texto, encontrarás un botón con el texto "Enviar". Utilizarás este botón para enviar el texto ingresado y obtener una sugerencia de letra.
5. Etiqueta de resultado: Debajo del botón "Enviar", hay una etiqueta vacía donde se mostrará el resultado de la sugerencia de letra.

Enviar el texto y obtener la sugerencia de letra

1. Escribe tu texto deseado en el campo de entrada de texto.
2. Haz clic en el botón "Enviar".
3. La aplicación analizará el texto ingresado y sugerirá la letra más repetida como recordatorio.
4. El resultado de la sugerencia de letra se mostrará en la etiqueta de resultado.

Cambiar la letra seleccionada (opcional)

Si deseas cambiar la letra seleccionada, puedes seguir estos pasos:

1. Asegúrate de haber recibido una sugerencia de letra en el paso anterior.
2. Verás la letra seleccionada en la parte inferior de la ventana, junto con la opción "Cambiar letra".
3. Haz clic en el botón "Cambiar letra".
4. Se abrirá una nueva sección donde podrás ingresar una nueva letra.

5. Escribe la nueva letra en el campo de entrada y haz clic en el botón "Aceptar".
6. La letra seleccionada se actualizará y se mostrará en la etiqueta correspondiente.

Confirmar la impresión

Si deseas confirmar la impresión de un archivo llamado "letra.gcode", sigue estos pasos:

1. Asegúrate de haber seleccionado una letra (ya sea sugerida o cambiada manualmente) en los pasos anteriores.
2. Haz clic en el botón "Imprimir".
3. Se abrirá un cuadro de diálogo que te permitirá seleccionar una carpeta para guardar

"Esperamos que que este manual de usuario haya sido útil y te haya proporcionado las instrucciones necesarias para utilizar nuestra Interfaz de Usuario.