

Pontificia Universidad Javeriana
Ingeniería de Sistemas
Estructuras de Datos

Juan Sebastián Mondragón
David López
William Gómez

Proyecto Entrega 1

Diseño: TAD's

TAD controlador_juego

Conjunto mínimo de datos

- diccionario, lista de cadena de caracteres, lista donde se almacena por valor una palabra proveniente del diccionario. Esta lista se encuentra en el mismo orden en el que se encuentra el archivo de entrada
- diccionario_inverso , lista de cadena de caracteres, lista donde se almacena por valor una palabra proveniente del diccionario. Esta lista se encuentra en orden reverso, es decir desde el final del archivo cargado hasta el inicio

Operaciones

- vLetras(letras), recibe una cadena de caracteres como parámetro, verifica que sea un tablero de juego válido (solo letras y que sean 7 de estas) y retorna verdadero o falso si se cumple dicha condición
- vComando(palabras), recibe como parámetro una lista de cadena de caracteres, los cuales son los textos ingresados por el usuario en el menú, separados por " " . Determina si el comando ingresado es válido, si tiene un segundo argumento, este podrá ejecutar la función del comando si esta cumple con las condiciones de ejecución, que dependen del comando.
- inicializarDiccionario(nombre archivo), recibe como parámetro una cadena de caracteres que representa el nombre del archivo, si este archivo existe se carga su contenido en la lista diccionario.
- inicializarDiccionarioInverso(nombre archivo), recibe como parámetro una cadena de caracteres que representa el nombre del archivo, si este archivo existe se carga su contenido en la lista diccionario_inverso junto con su debido orden de carga.
- puntajePalabra(palabra), recibe como parámetro una cadena de caracteres que representa la palabra a evaluar, verifica si hay algún diccionario cargado, verifica si la palabra ingresada existe en algún diccionario, y retorna el puntaje de la palabra utilizando la operación calcularPuntaje del TAD puntaje

TAD puntaje

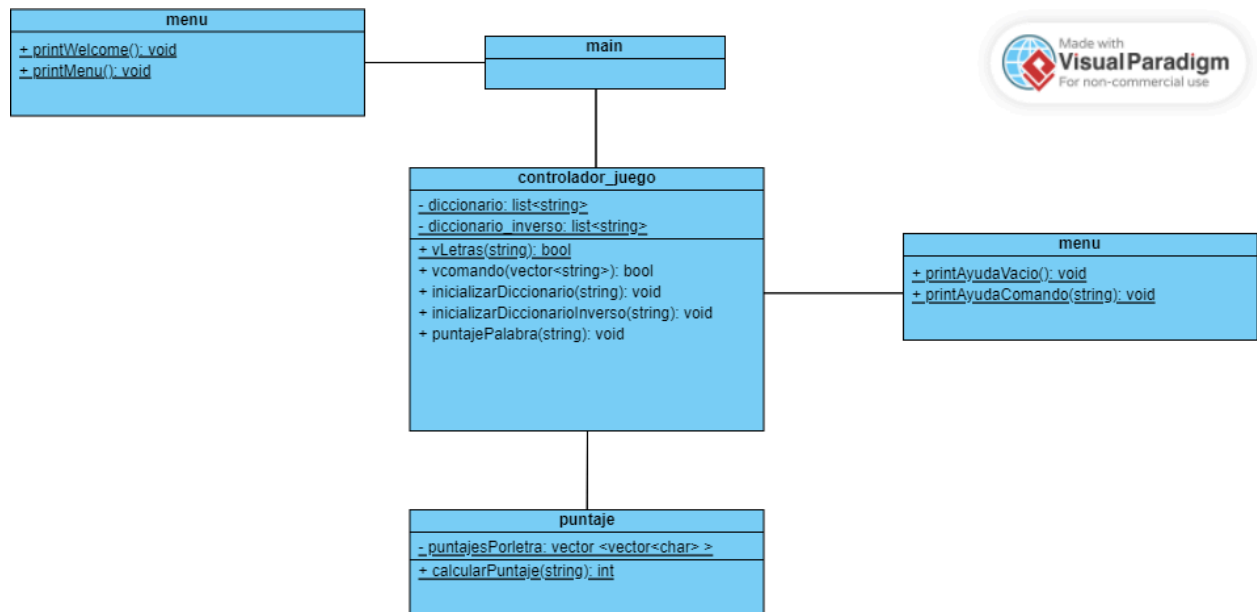
Conjunto mínimo de datos

- puntajesPorletra, lista de lista de caracteres, lista donde se almacenan las letras y su valor de puntaje dentro del juego. Cada sub lista tiene su puntaje asignado y las letras que tienen dicho puntaje.

Operaciones

- puntaje(), constructor de la clase, se crea la lista puntajesPorletra con los mismos datos siempre que se crea.
- calcularPuntaje (palabra), recibe una palabra de parámetro, revisa con el dato puntajesPorletra cada carácter de la palabra y retorna la suma de los puntajes de todos los caracteres de la palabra

Diseño: diagrama de relación



Diseño: inicializar diccionario

1. Ingresar comando inicializar junto con el nombre del archivo donde se contiene el diccionario
2. Se verifica si el archivo existe, y que la lista de diccionario se encuentre vacía, de ser así se abre el archivo y se comienza a leer línea por línea, almacenando cada dato en la cola de la lista, para tenerla en orden.
3. Verificar que la lista no quede vacía después del proceso de lectura
4. De ser exitoso, confirmar al usuario que se ha inicializado con éxito

Diseño: inicializar diccionario inverso

1. Ingresar comando iniciar inverso junto con el nombre del archivo donde se contiene el diccionario
2. Se verifica si el archivo existe, y que la lista de diccionario inverso se encuentre vacía, de ser así se abre el archivo y se comienza a leer línea por línea, almacenando cada dato en la cola de una lista temporal, para tenerla en orden.
3. Se recorre la lista temporal al revés y se almacenan los datos en la lista diccionario inverso, de manera que esta contenga los datos al revés de como están en el archivo de entrada
4. Verificar que la lista no quede vacía después del proceso de lectura
5. De ser exitoso, confirmar al usuario que se ha inicializado con éxito

Diseño: puntaje

1. Ingresar comando puntaje junto con la palabra que se quiere evaluar
2. Se verifica que por lo menos alguno de los dos diccionarios se encuentren inicializados (no vacíos), de ser así, se verifica que la palabra ingresada exista en alguno de los diccionarios.
3. Se inicializa la multilista donde se almacenan las letras con su puntaje respectivo (tanto mayúsculas como minúsculas)
4. Se crea un entero que almacena el total del puntaje, se compara letra a letra de la palabra ingresada con dicha multilista para determinar el puntaje POR letra, se suma el valor de cada letra ese entero
5. Retorna el entero y le indica al usuario cual es el valor del puntaje de la palabra ingresada

Diagrama: inicializar diccionario

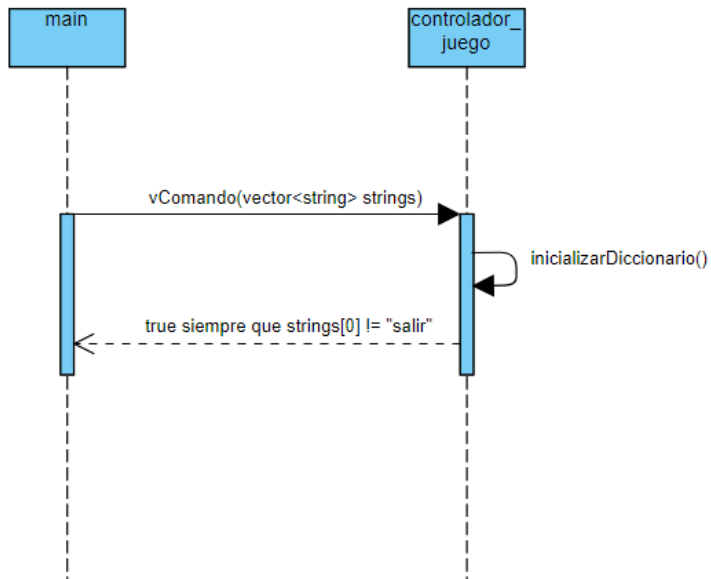


Diagrama: inicializar diccionario inverso

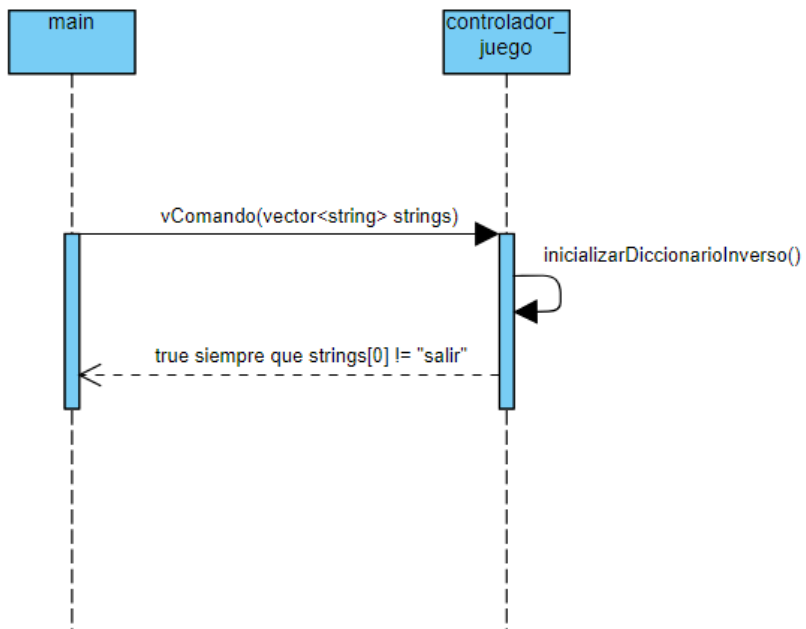
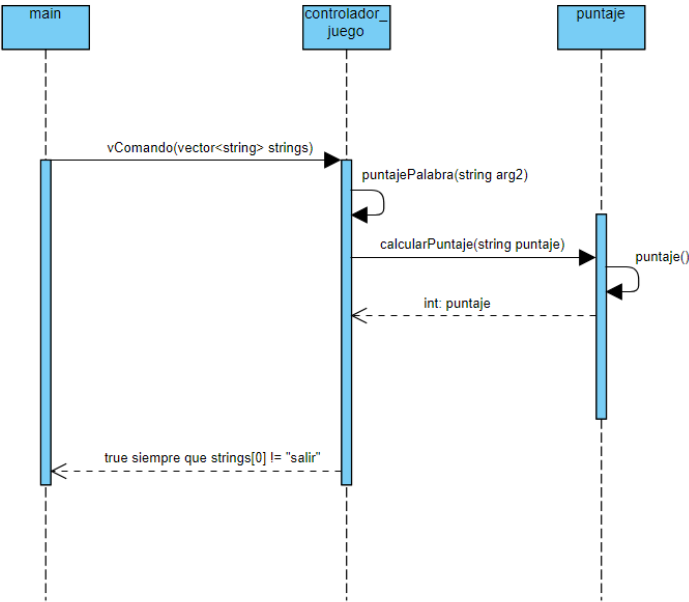


Diagrama: puntaje



Plan de pruebas

Pruebas de diccionario:

- Ingresar nombres de archivos no válidos, el sistema debería de retornar el mensaje de error.
- Ingresar nombres de archivos que ya fueron inicializados, el sistema debería de retornar un mensaje de que ya se encuentra cargado.
- Ingresar nombres de archivos que existen y no han sido inicializados, el sistema debería de retornar un mensaje de éxito.

Pruebas de usuario:

- Ingresar un tablero invalido (de más de 7 letras o con caracteres inválidos), el sistema debería de retornar un mensaje de error
- Ingresar un comando invalido, el sistema debería de retornar un mensaje de error y pedirle al usuario ingresar un nuevo comando
- Ingresar un comando sin su segundo argumento, el sistema debería de retornar un mensaje de como usar el comando y pedirle al usuario ingresar un nuevo comando.
- Ingresar un comando con un segundo argumento invalido, el sistema debería de retornar un mensaje de error y pedirle al usuario ingresar un nuevo comando.
- Ingresar un comando con su segundo argumento válido, el sistema deberá de realizar la acción que el usuario invoca a través del comando
 - Comando inicializar e iniciar_inverso:
 - Si el segundo argumento es un archivo válido, el sistema debería de retornar el mensaje de diccionario inicializado con éxito.
 - Si el segundo argumento es un archivo invalido, el sistema debería de retornar un mensaje de error y pedirle al usuario ingresar un nuevo comando
 - Si el segundo argumento es un archivo válido pero el diccionario ya se encuentra inicializado, el sistema debería de retornar el mensaje de que el diccionario ya se encuentra inicializado y debería pedirle al usuario ingresar un nuevo comando
 - Comando puntaje
 - Si no hay diccionario cargado el sistema debería de retornar un mensaje de error donde se dice que no hay diccionario inicializado, y debería pedirle al usuario ingresar un nuevo comando
 - Si la palabra ingresada no existe en un diccionario inicializado, el sistema debería de retornar mensaje de error de palabra no encontrada en los diccionarios y debería pedirle al usuario ingresar un nuevo comando
 - Si la palabra ingresada existe en un diccionario inicializado, el sistema debería de retornar un mensaje con el puntaje de la palabra