

Laboratorio 1: Reto Trivia

Mariana Sierra López y Angel Albeiro Ricardo Ruiz

Resumen— En el presente documento se evidenciará la realización de la práctica de laboratorio que se basa en la continuación de la realización de un juego de Preguntados (Trivia Crack), haciendo uso de los archivos, métodos de archivos y funciones. Se espera que al realizar las funciones el servidor funcione correctamente con el documento “users.py”
palabras claves: archivos, funciones, servidor

I. INTRODUCCIÓN

Para realizar esta práctica se suministraron cuatro archivos, donde cada archivo cuenta con una función distinta.

Los archivos “trivia_server.py” y “trivia_client.py” son archivos que no se pueden modificar ya que cuentan con la funcionalidad básica de la conexión del servidor y la operación de este en una red local o en el mismo equipo de prueba. El archivo “test_trivia_client.py” es un archivo que se puede modificar para verificar el funcionamiento del servidor.

Se buscó lograr que el código del servidor funcione correctamente, para poder generar una interconexión en distintos dispositivos, para esto el propósito de la práctica es crear cierto tipo de funciones especificadas en “user.py”, que son la base para el correcto funcionamiento del juego que se espera lograr al final de semestre. En la práctica se logró generar las funciones requeridas utilizando archivos y métodos de este, tales como “split()”, “strip(“,”)”, “write()”, “readlines()” y en el manejo de archivos se usó “r”, “w” y “a+” con el fin de que el cliente se encuentre registrado, cierre su sesión y actualice y muestre su puntaje. Para comprobar estas funciones lo que se hizo fue comprobar cada una por separado.

II. PROCEDIMIENTOS

Para iniciar la práctica de laboratorio lo primero que se hizo fue instalar el “pip install request” para poder realizar un

correcto funcionamiento.

Lo siguiente fue la creación de los códigos de las funciones necesarias para la práctica, ya que estas funciones son un requerimiento para el proyecto final.

diagramas de flujo importantes:

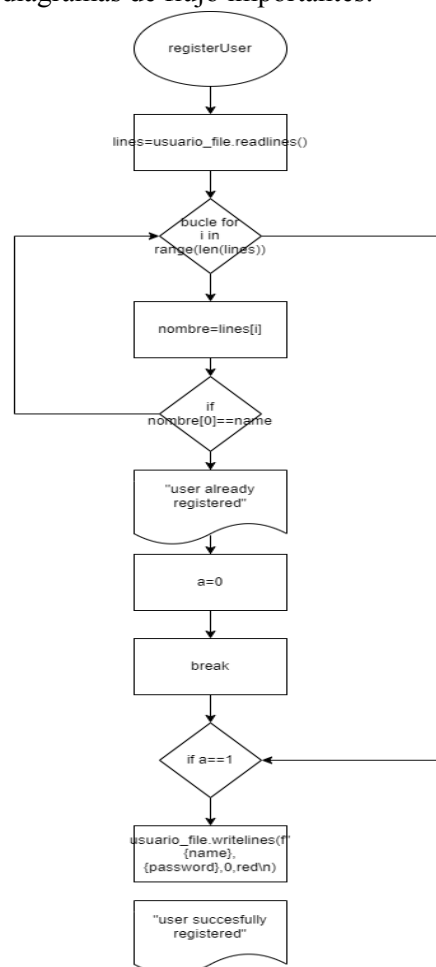


figura 1:diagrama de flujo de función para registrar usuario

en la primera función se abre un archivo en el cual se va a guardar los diferentes usuarios que van a estar conectados, para esto se abre el archivo y se guardan las líneas del archivo en una variable, lo siguiente es utilizar un bucle “for” para guardar una línea específica

en otra variable, pero la forma de guardarla como una variable de tipo lista, y luego hacer la comparación si el nombre que se está introduciendo ya está en el archivo, esto para saber si ya está registrado o para registrarlo.

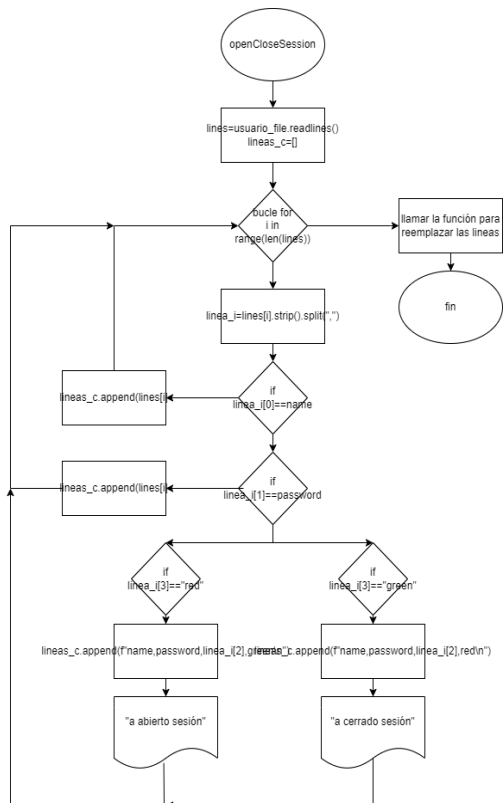


figura 2:diagrama de flujo para abrir o cerrar sesión

Para abrir o cerrar sesión se abre el archivo en el que están los usuarios registrados, se guarda en una variable las líneas del archivo y después en un bucle “for” se guarda una de las líneas en una variable tipo lista, luego se hace la comparación si el nombre dado está en esa línea, si la contraseña dada está en la misma línea y se comprueba si el usuario está conectado o no, para cambiar su estado, para comprobar esto en el archivo se guardó una variable que comprueba el estado y así realizar la comparación.

En la función “updateScore” su funcionamiento es muy parecido a la función para abrir o cerrar sesión, el único cambio significativo es que no se reemplaza el estado del usuario en el archivo, si no que se

reemplaza el nuevo puntaje, pero sigue realizando las demás comparaciones.

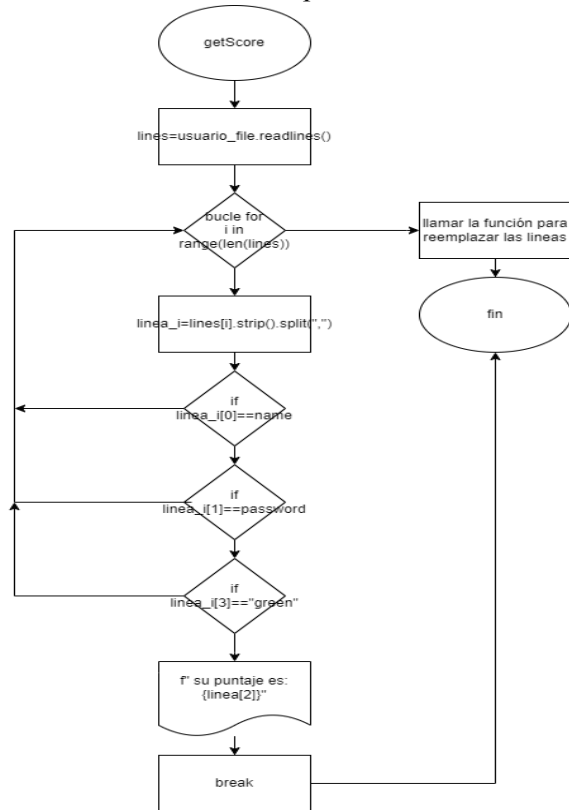


figura 3:diagrama de flujo de la función para mostrar el puntaje

la función para mostrar el puntaje es bastante más sencilla que las anteriores, esta función abre el archivo donde se registran los usuarios y guarda todas las líneas del archivo en una variable, luego, con el bucle “for” guarda una línea específica en una variable tipo lista y hace la comparación para comprobar si el nombre, contraseña están registrados y también comprueba el estado del usuario, si todas las comparaciones son correctas va a mostrar el puntaje del usuario que se está comparando.

Para la función “usersList” se codifica de manera muy parecida pero al momento de hacer la comparación si el usuario está conectado, se crea un nuevo bucle for que comprueba que usuarios están conectados y los muestra en pantalla con su puntaje

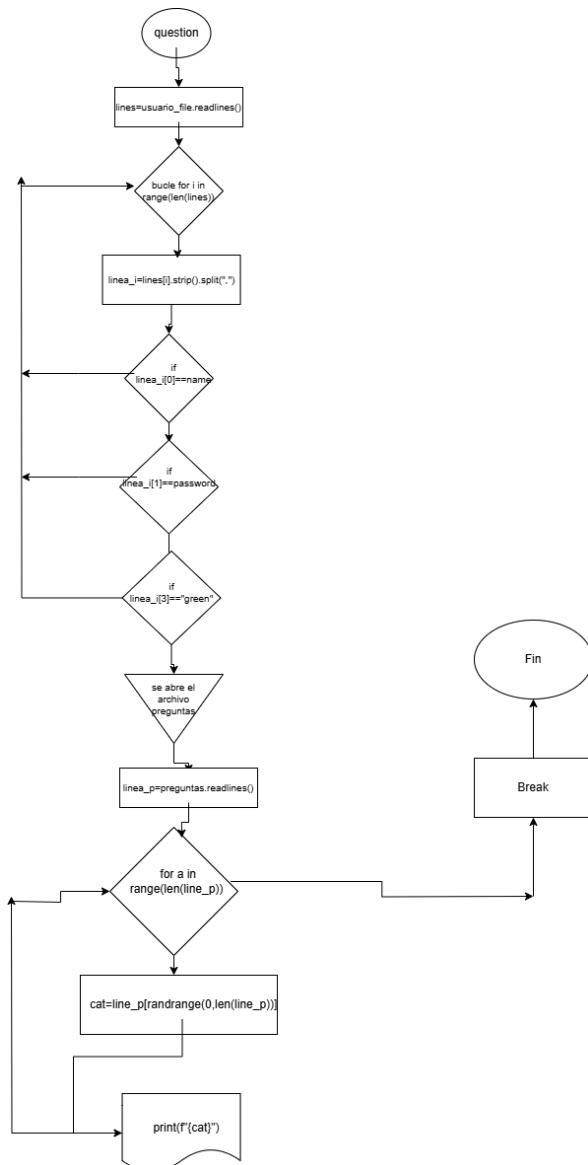


Figura 4: diagrama de flujo de la función para mostrar los usuarios conectados.

Para la función de mostrar una pregunta hace prácticamente lo mismo que en los casos anteriores, abrir el archivo, guardar las líneas en una variable, guardar una línea específica como una lista en una variable y si se cumplen las comparaciones del nombre, contraseña y el estado del usuario es activo, muestra el nombre del usuario y su puntaje

Para la función de mostrar pregunta realiza el mismo procedimiento anterior guardar las

III. ANÁLISIS Y RESULTADOS

Objetivo:

Se esperaba lograr el correcto funcionamiento de las funciones a crear y que los códigos se comuniquen de manera correcta entre sí.

Resultado:

Se lograron crear las funciones de manera correcta usando archivos y bucles condicionales, esto fue un resultado positivo, en cuanto a la comunicación entre los códigos, no se obtuvo el mejor resultado, ya que el código del servidor al momento de comunicarse con el código “users” no lo hace de la manera que se espera.

Acciones:

Para la obtención de los resultados ya mencionados, se hizo una investigación sobre el uso de archivos, como los modos de abrir y el cómo poder modificar el contenido de estos, además, de también centrarse en un correcto uso de los bucles condicionales para lograr una óptima codificación de los códigos de las funciones.

No se lograron obtener todos los resultados esperados por la falta de conocimiento en el funcionamiento del código del servidor, de esta manera no se logran todos los objetivos.

Objetivos a futuro:

Poder lograr la correcta comunicación entre los códigos para que el servidor funcione de manera correcta, además, de mejorar la redacción en las funciones utilizadas.

Estrategias:

Buscar e indagar la manera de que estos códigos se comuniquen correctamente entre sí, investigar y consultar técnicas para la próxima práctica de laboratorio para una mejor codificación de archivos.