**TESTING DE APLICACIONES**

CONTROL DE CALIDAD

INTEGRANTES:

Barreto Maximiliano

Bulacio Iara

Calle Castro Alejandra

Duarte Nicolás

Herrera Malena

Jara Fernanda

Tejerina Mirta

INSTITUTO DE FORMACION PROFESIONAL 403

PROFESORA : Fabiana Silva

**28 /08/2024**

**Introducción**

Los controles de calidad son ideales a la hora de testear distintas aplicaciones,abarca una variedad de técnicas y metodologías diseñadas a fin de identificar fallos y corregirlos, a su vez mejora el equilibrio de una aplicación garantizando que el producto cumpla con los estándares esperados por el público.

En el presente trabajo se buscó realizar diferentes pruebas de calidad a un programa de código abierto, esperando así poder ubicar diversas fallas, por lo tanto se optó por la aplicación “WINE”, la cual es de código abierto , cuya funcionalidad era hacer compatibles en un el sistema operativo linux aplicaciones solo funcionales en windows

No obstante,no se realizó dicho testeo porque se presentaron complicaciones durante la ejecución de la misma (la problemática se desarrollará a continuación), por lo cual se decidió cambiar la aplicación por“SPEED DREAMS 2.3.0” un videojuego simulador de carreras.

**Desarrollo**

Rol de cada integrante del grupo:

Fernanda Jara : Test de usabilidad y redacción del informe.

Malena Herrera :Test de usabilidad e investigación.

Iara Bulacio :Test de usabilidad y redacción del informe.

Mirta Tejerina :Informe e investigación sobre normas APA.

Alejandra Calle Castro :Líder,Investigación y redacción de informe.

Nicolas Duarte: Test de usabilidad y automatización.

Maximiliano Barreto :Test de usabilidad y propuesta de programas.

La problematica principal fue que wine requeria de Linux para funcionar,sistema operativo que los miembros del equipo no manejaban y para instalarlo se requería hacer un particionado de disco y generar un dual boot(particionado: decirle al disco que tanta cantidad de gb es para windows y tanta cantidad de gb es para Linux.Dual boot: del inglés arranque dual,te permite elegir cuando se inicia la computadora entre usar linux o windows) lo que ponia en riesgo sus respectivos discos por corrupcion de datos,en su momento se considero usar una maquina virtual,pero luego de ver la complejidad en el uso de la consola de comandos y carencia de fallos de aplicaciones compatibles se optó por descartar la opcion de wine

Pruebas de compatibilidad usando wine en computadora de la escuela con Linux y pruebas en windows sin wine:

* Gimp: un programa de codigo abierto editor de imágenes que se asemeja a Photoshop pero con distintas funcionalidades:
  + No sé dectaron problemas en la ejecución,se requieren conocimientos avanzados en diseño grafico para continuar
* Calibre: es un programa administrativo de codigo abierto que busca organizar todo tipo de documentos en un entorno gráfico para que el usuario tenga fácil acceso a ellos

Problemas encontrados:

* Problemas estéticos: su aspecto se ve muy antiguo

Resultados:

El grupo considero que las pruebas realizadas eran insuficientes para generar un informe detallado de fallos y posterior propuesta de soluciones,ya que los programas testeados estaban muy completos,se propuso un nuevo progrmaa para testear

Speed dreams 2.3.0

Speed dreams 2.3.0 es un juego simulador de codigo abierto escrito en (agregar lenguajes,son varios) y que usa el motor gráfico (agregar motor gráfico) se considero que al ser un juego en desarrollo serian mas detectables las fallas originadas por su uso,a continuación se detallan las fallas encontradas tanto en pruebas manuales como automatizadas:

\*Agregar fallas encontradas

La problemática principal fue que al intentar abrir el código para hacer las pruebas correspondientes se necesitó descargar Ubuntu, lo cual generó muchas dificultades ,por lo tanto luego se probó con el programa “GIMP” y se obtuvo la misma falla (Se debía descargar Ubuntu para poder testear ese programa).Por ello, se tomó la decisión de cambiar al videojuego llamado “SPEED DREAMS 2.3.0” cuya función es simular carreras de autos en un circuito..

En principio, pensamos instalar en Python la librería “Pygame” o “Playwright” ya que investigamos que ambas eran una herramienta sencilla para crear y automatizar pruebas en videojuegos, para ello es necesario en primer lugar instalar “pip”, el gestor de paquetes de Python. Se intentó por Windows pero no se ejecutaba el comando “pip install”, entonces se optó por instalarlo utilizando la consola de Linux a través de powershell, sin embargo, la misma requiere una clave que estaba fuera de nuestro alcance. Finalmente, se desistió en la idea, no solo por el fallo técnico, sino también porque el lenguaje en el que está programado el videojuego es en C++ y no contamos con el tiempo y los conocimientos necesarios para traducirlo a Python o bien añadir modificaciones al código original que resuelvan las fallas.

Se investigó sobre ambas opciones de testeo, (tanto manual como automatizada) y se descubrió que al automatizar pruebas en el programa “SPEED DREAMS 3.2.0” este era muy complejo ya que el juego contenía el

Nico está fabricando un programa de testeo automatizado

código con el lenguaje C + +.

Desafortunadamente la aplicación excede los conocimientos actuales de cada integrante y debido a la falta de tiempo se decidió realizar pruebas manuales únicamente.

A continuación se presentan las múltiples fallas que se encontraron durante el proceso.

1. Al intentar abrir la aplicación, se encontró fallos en la carga del juego, es decir que se tilda, lo cual puede ser una posible mala gestión de memoria..
2. El usuario pierde el control del vehículo con facilidad, lo que causa que el automóvil se deslice fuera de la pista. Esto interrumpe la continuidad del juego y afecta negativamente la jugabilidad, impidiendo una experiencia fluida y satisfactoria.
3. En el sector de la configuración de player, se complica a la hora de querer modificar algún detalle,como por ejemplo el nombre que desees elegir como jugador,(al menos que presiones "cancel" y repetir la acción nuevamente) .
4. La Resolución máxima no se adapta a los monitores, lo cual puede generar fallos en el mismo.
5. La cámara aérea a veces genera niebla, se ve borroso y carga mal los sprites.
6. En el mapa mantón speedway usando bots simplix entra a los pits cuando no debería.
7. Hay dos páginas del juego con distintos dominios.
8. La página tiene desperfectos con php, se ve desfasado.
9. Foro en desuso.
10. El juego es detectado como malicioso por falta de certificados.
11. Los carteles no tienen comisiones.
12. Aparece el CMD, es bueno para poder testear pero a su vez es malo estéticamente para la versión final.
13. De noche no hay buena resolución de imagen.

Afortunadamente al hacer una puesta en común se encontraron posibles soluciones a los fallos encontrados en el videojuego (mencionados anteriormente).

Muchas de la soluciones encontradas son las siguientes:

Configurar para que no aparezca el CMD al iniciar el programa.,Mejorar la física para tener una experiencia más realista,como también las interfaces y la página para reactivar el foro para que los usuarios puedan escribir sus experiencias y poder calificar la aplicación.

Usar herramientas oficiales para que el sistema no lo declare como virus y lo tomé como un programa de uso legal.

Implementar una mejora en los autos para que se dañen. Renderizar los frames por segundo (FPS),para aumentar la fluidez del programa. Reprogramar la parte de los controles para incrementar la satisfacción del usuario.Arreglar la base de datos donde almacena los datos del jugador para que en vez borrar la información , la guarde para futuro uso.

dejaba llenar información. Reajustar la calidad y los ángulos de la cámara

posible error de gestión de memoria. Reprogramar la inteligencia artificial de los bots

posible error de gestión de memoria.Reconfigurar resolución de las interfaces.

**Conclusión**

Para finalizar,como se ha demostrado durante el informe el objetivo principal fue mostrar el proceso de control de calidad.Cada participante se desempeñó en el rol decidido por la líder para así poder contribuir en dicho informe.

(Jorge Paz, 2022)

(A Tester Mate, 2020)

(Geek QA, 2022)

(Gluo, 2023)

(Juan Luis Restituyo, 2023)

(J2logo, 2021)

(Geek QA, 2021)

(Quality-Stream, 2019)

(Pythonista, 2022)

(Johanna Rodríguez, 2021)

(Investigar es fácil, 2022)

(Massarik, 2020)

**Referencias**

A Tester Mate. (10 de nov de 2020). *Fundamentos de Pruebas - Tipos y Niveles de Pruebas de Software.* Obtenido de https://www.youtube.com/watch?v=V1

Geek QA. (12 de dic de 2021). *Automatización de pruebas con Selenium IDE | Curso paso a paso con ejemplo🔥*. Obtenido de https://www.youtube.com/watch?v=KskSQI5ZPVM&t=483s

Geek QA. (28 de mar de 2022). *Practica Software Testing con estos sitios✅ | NO BUSQUES MÁS...* Obtenido de https://www.youtube.com/watch?v=i7

Gluo. (14 de mar de 2023). *Pruebas de usabilidad: ¿Qué son y cómo realizarlas?*

Gluo. (14 de mar de 2023). *Pruebas de usabilidad: ¿Qué son y cómo realizarlas?* Obtenido de https://www.youtube.com/watch?v=AnwYsYD5\_3c

Investigar es fácil. (23 de abr de 2022). *Cómo referenciar un video de YouTube con Normas APA 7ma edición- Investigar es fácil*. Obtenido de https://www.youtube.com/watch?v=0TZW1lnEZZQ

J2logo. (5 de feb de 2021). *TESTS UNITARIOS en Python (Introducción a las 🚀 Pruebas de código)*. Obtenido de https://www.youtube.com/watch?v=0C24K0KlWXU

Johanna Rodríguez. (19 de feb de 2021). *FORMATO 📚NORMAS APA SÉPTIMA EDICIÓN WORD📘*. Obtenido de https://www.youtube.com/watch?v=Akx0kH57Q80&t=11s

Jorge Paz. (14 de enero de 2022). *Niveles de Prueba, Tipos de Pruebas y Clases de Prueba*. Recuperado el 15 de agosto de 2024, de https://www.youtube.com/watch?v=yP

Juan Luis Restituyo. (19 de jun de 2023). *Pruebas de Usabilidad*. Obtenido de https://www.youtube.com/watch?v=4OvTDueifA0

Massarik. (30 de ene de 2020). *CÓMO CITAR UN VIDEO DE YOUTUBE EN WORD CON EJEMPLO SEGÚN NORMAS APA SÉPTIMA EDICIÓN (7ma.)*. Obtenido de https://www.youtube.com/watch?v=Ee42ORf5U0E

Pythonista. (12 de ene de 2022). *py411 Automatización de pruebas con Python, Selenium y Postman 220111*. Obtenido de https://www.youtube.com/watch?v=RUQQXdjxhkY&t=158s

Quality-Stream. (28 de ene de 2019). *Introducción a la Automatización de Pruebas de Software*. Obtenido de https://www.youtube.com/watch?v=R\_hh3jAqn8M