**Orientación Orientada A Objetos**

**Excepciones**

**Octubre 2017**

**Laboratorio 4/6**

**Andrés Felipe Dávila Gutiérrez Y Javier Esteban Lopez Peña**

**EQUIPOS**

EN BLUEJ

PRACTICANDO MDD y BDD con EXCEPCIONES

En este punto vamos a aprender a diseñar, codificar y probar usando excepciones. Para esto se

van a trabajar dos métodos de la clase Equipo y la excepción EquipoExcepcion.

**1. En su directorio descarguen los archivos contenidos en equipo.zip, revisen el contenido**

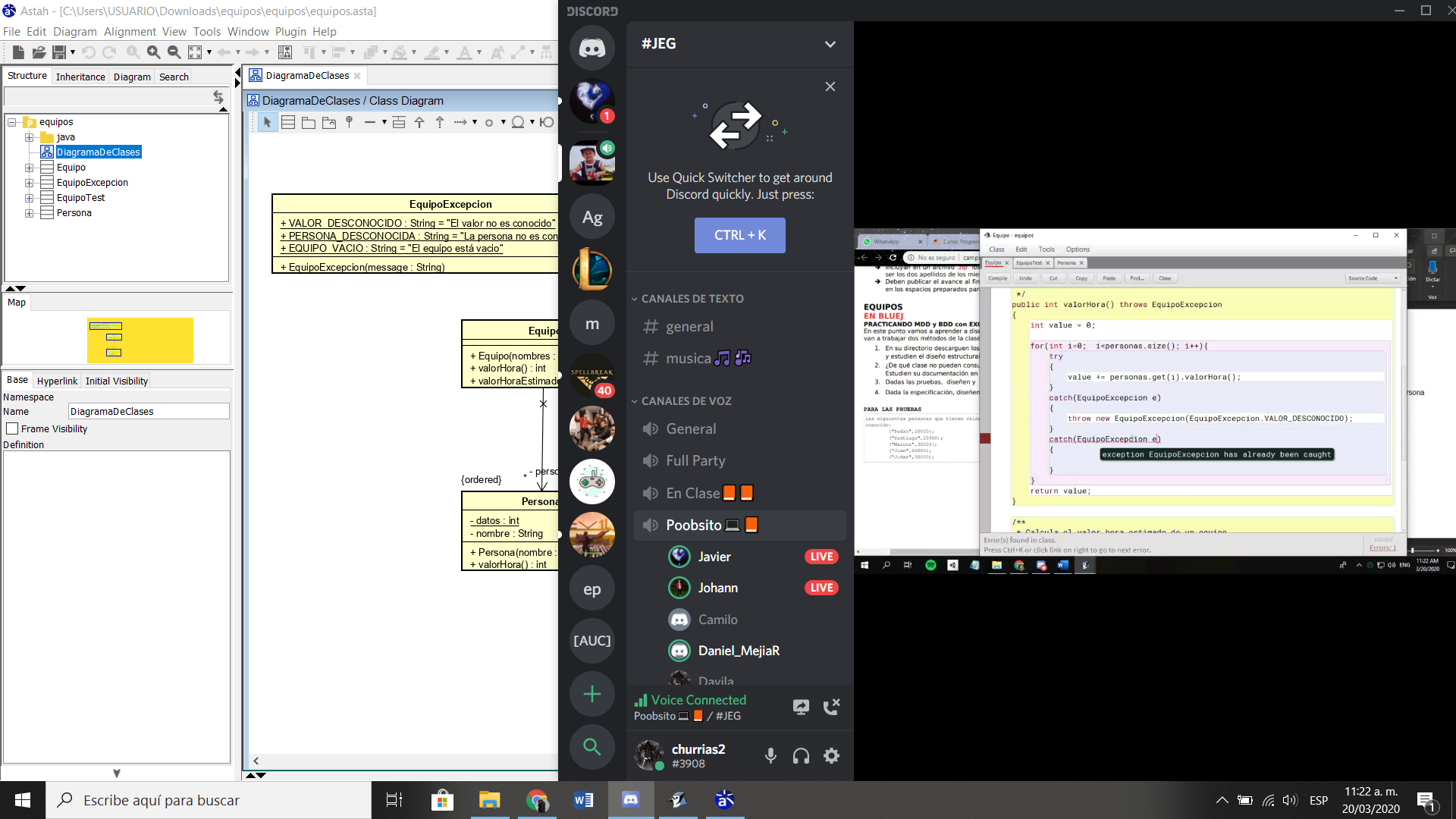
**y estudien el diseño estructural de la aplicación.**

**2. ¿De qué clase no pueden consultar las fuentes? ¿En qué formato está? ¿Por qué?**

Estudien su documentación en el diseño o en el código (/doc).

De la clase personas no podemos consultar las fuentes, están en formato de HTML, no se puede consultar porque esta eliminado el archivo dentro del paquete que nos dan.

**3. Dadas las pruebas, diseñen y codifiquen el método valorHora.**

**4. Dada la especificación, diseñen, codifiquen y prueben el método valorHoraEstimado**.

**SINAP. Sistema Nacional de Áreas Protegidas**

**EN CONSOLA**

**Conociendo el proyecto SINAP (No olviden respetar los directorios bin docs src)**

**1. En su directorio descarguen los archivos contenidos en SINAP.zip, revisen el contenido. ¿Cuántos archivos se tienen? ¿Cómo están organizados? ¿Cómo deberían estar organizados?**

Vienen 5 archivos .java y un archivo .asta, están organizados todos en una sola carpeta. Deberían existir carpetas bin, docs y src para cuando los archivos .class y .txt esten

**2. Estudien el diseño del programa: diagramas de paquetes y de clases. ¿cuántos paquetes tenemos? ¿cuántas clases tiene el sistema? ¿cómo están organizadas? ¿cuál es la clase ejecutiva?**

Hay dos paquetes aplicación y presentación, en aplicación estan las clases Area y Sinap y en presentación esta la clase SinapGUI. Teniendo un total de 3 clases en el sistema, siendo SinapGUI la clase ejecutiva.

**3. Prepare los directorios necesarios para ejecutar el proyecto. ¿qué estructura debe tener? ¿qué instrucciones debe dar para ejecutarlo?**

Mkdir bin, mkdir src y mkdir docs. Para crear las .class compilo SinapGUI con javac -d bin src\aplicacion\\*.java src\presentacion\SinapGUI.java

**4. Ejecute el proyecto, ¿qué funcionalidades ofrece? ¿cuáles funcionan? Realicen el diagrama de casos de uso correspondiente.**

java -cp bin presentacion.SinapGUI. Listar, adicionar y buscar son las funcionalidades activas

**5. ¿De dónde salen las áreas iniciales? Revisen el código y la documentación del proyecto. ¿Qué clase pide que se adicionen? ¿Qué clase los adiciona?**

De sinap adicioneCinco(), acción adicionar de sinapGui

**Adicionar y listar**

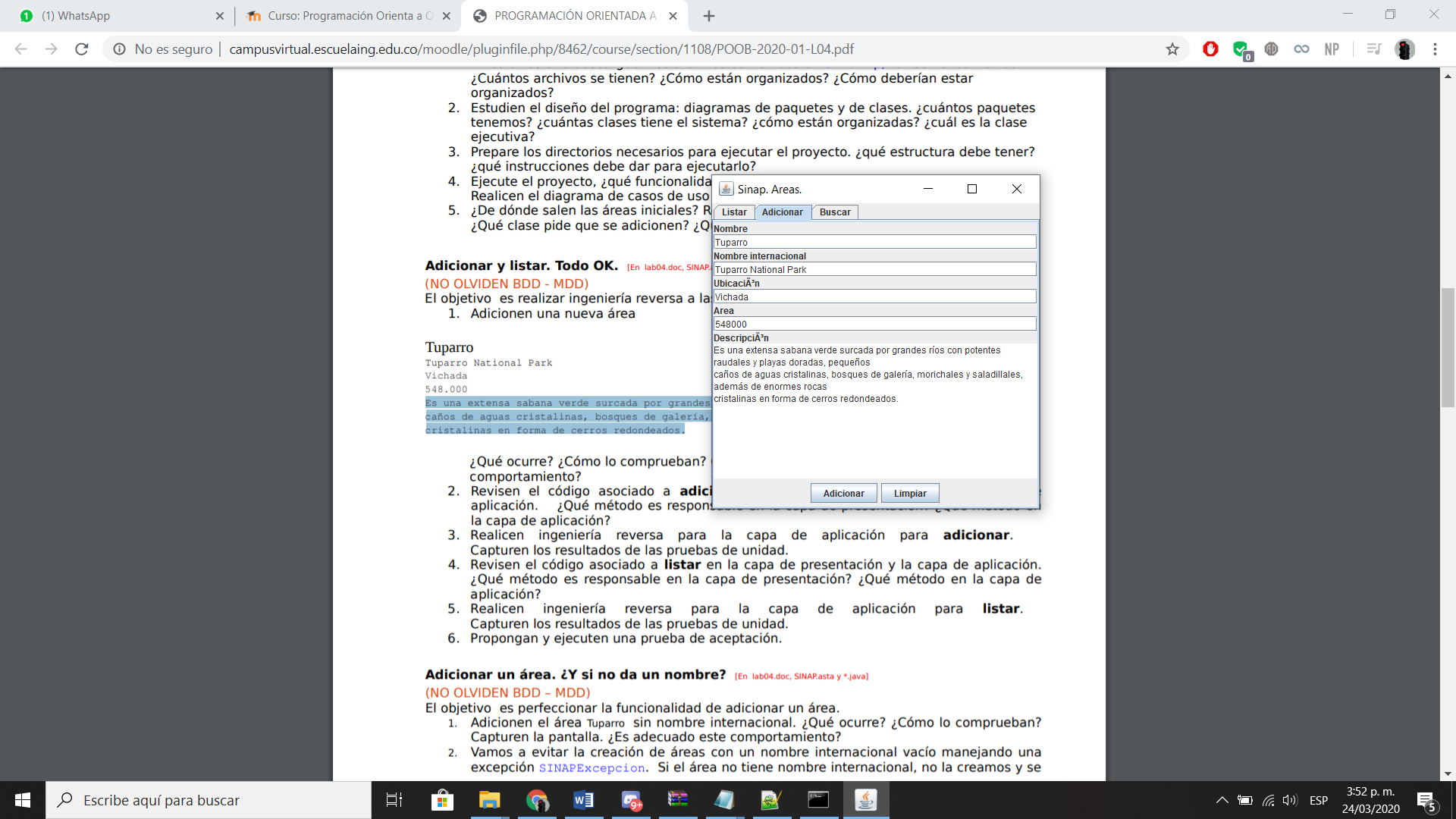
**El objetivo es realizar ingeniería reversa a las funciones de adicionar y listar.**

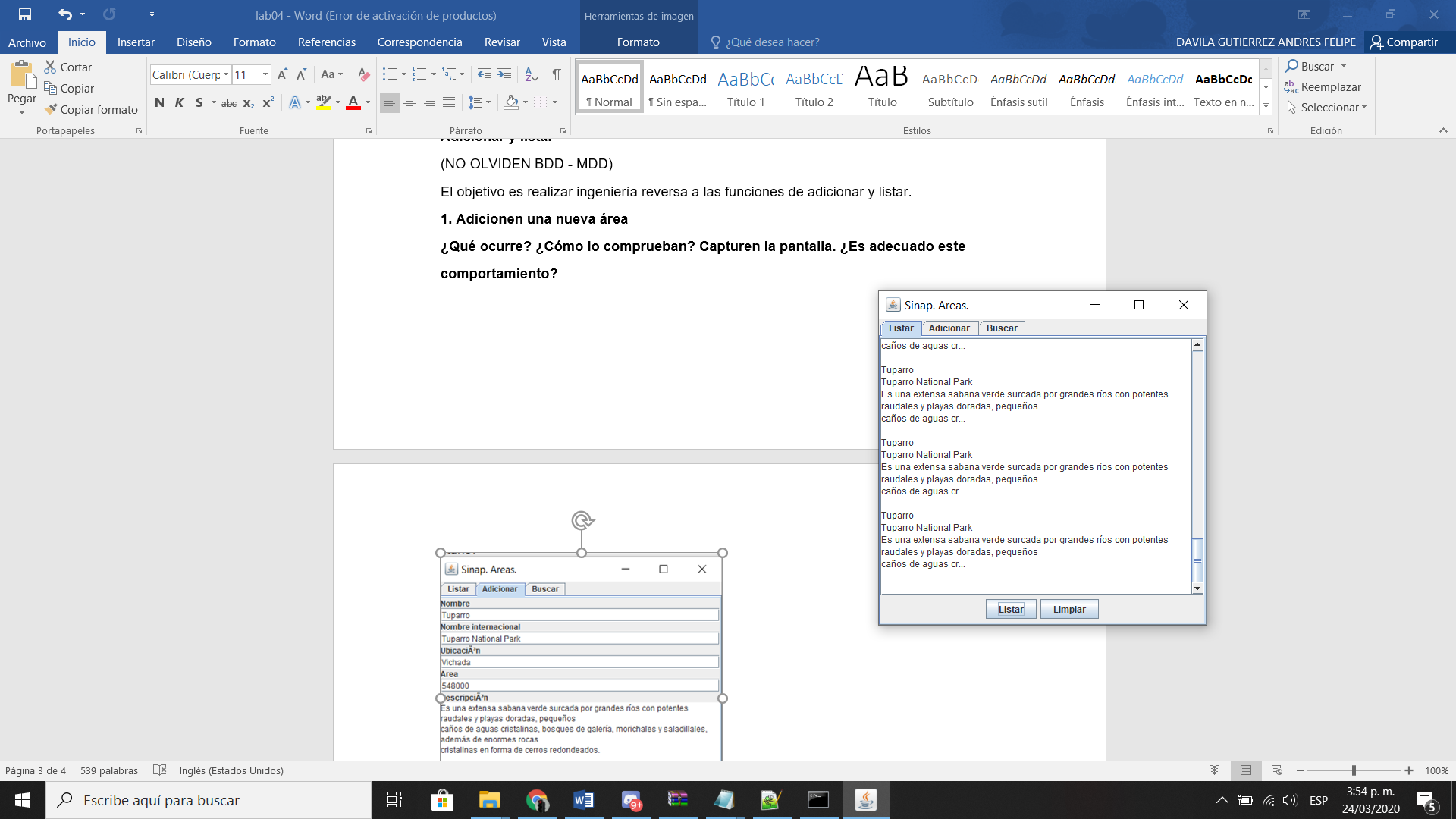
**1. Adicionen una nueva área**

**¿Qué ocurre? ¿Cómo lo comprueban? Capturen la pantalla. ¿Es adecuado este**

**comportamiento?**

En las imágenes se puede observan como se añade un área y lo podemos comprobar en listar,salen todas las áreas de sinap salen incluida la nueva que añadimos. El comportamiento es correcto





**2. Revisen el código asociado a adicionar en la capa de presentación y la capa de aplicación. ¿Qué método es responsable en la capa de presentación? ¿Qué método en la capa de aplicación?**

El método asociado en la capa de presentación es el método accionAdicionar u el método adicionar que se encuentra en sinap que recibe los parámetros y crea una nueva area

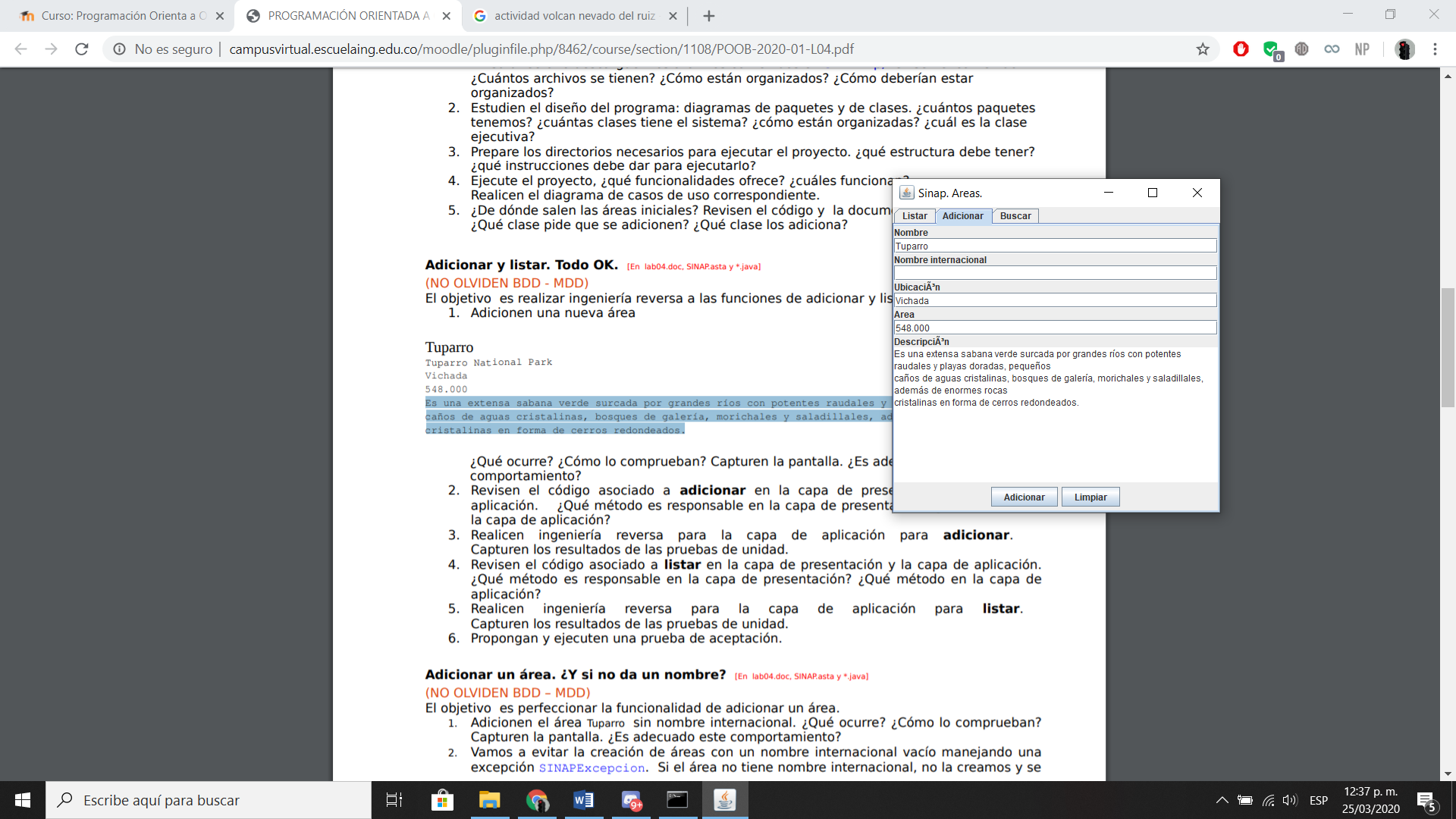
**3. Realicen ingeniería reversa para la capa de aplicación para adicionar. Capturen los resultados de las pruebas de unidad.**

**4. Revisen el código asociado a listar en la capa de presentación y la capa de aplicación. ¿Qué método es responsable en la capa de presentación? ¿Qué método en la capa de aplicación?**

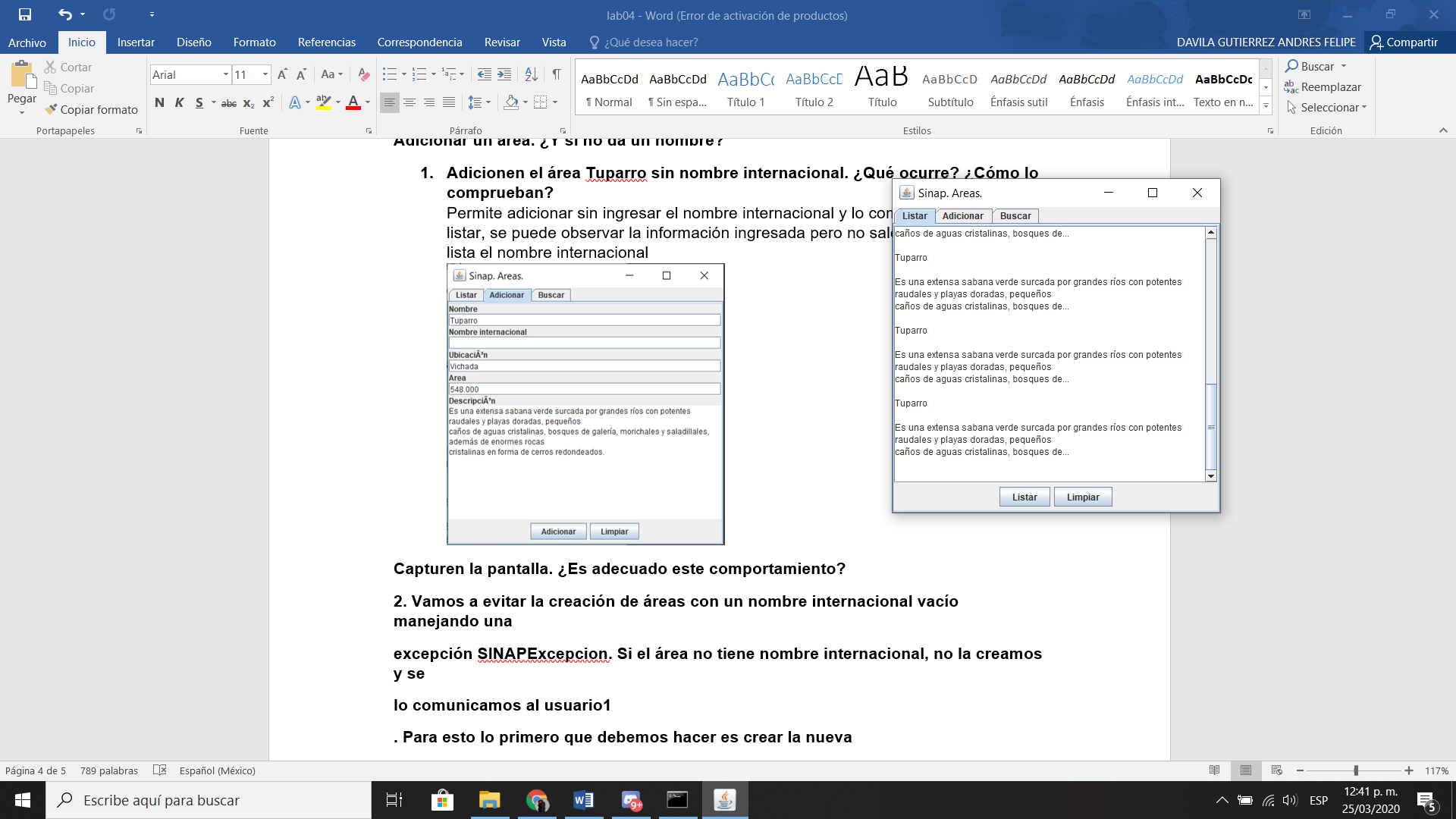
El método asociado en la capa depresentacion para listar es el método acción Listar y el método asociado a la capa de aplicación es el método toString

**5. Realicen ingeniería reversa para la capa de aplicación para listar. Capturen los resultados de las pruebas de unidad.**

**6. Propongan y ejecuten una prueba de aceptación.Adicionar un área. ¿Y si no da un nombre? uAdicionen el área Tuparro sin nombre internacional. ¿Qué ocurre? ¿Cómo lo comprueban?**

Permite adicionar sin ingresar el nombre internacional y lo comprobamos en el listar, se puede observar la información ingresada pero no sale en el espacio de lista el nombre internacional

**Capturen la pantalla. ¿Es adecuado este comportamiento?**

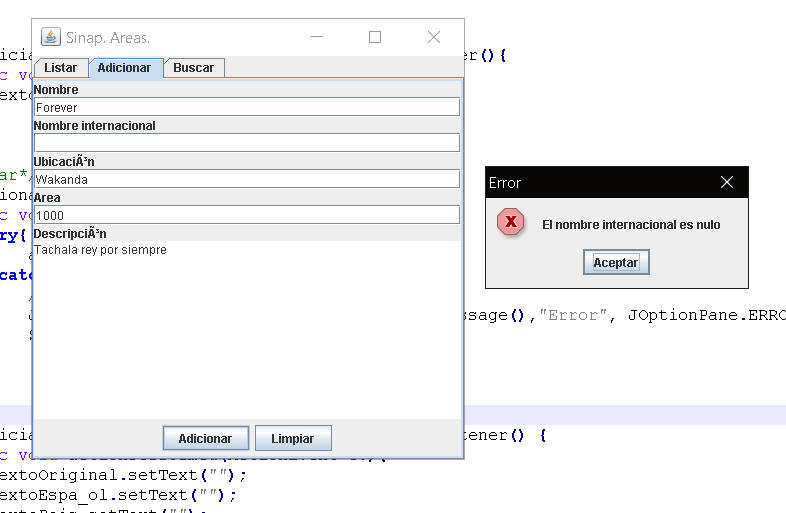


**2. Vamos a evitar la creación de áreas con un nombre internacional vacío manejando una excepción SINAPExcepcion. Si el área no tiene nombre internacional, no la creamos y se lo comunicamos al usuario1. Para esto lo primero que debemos hacer es crear la nueva clase SINAPExcepcion considerando este primer mensaje.**

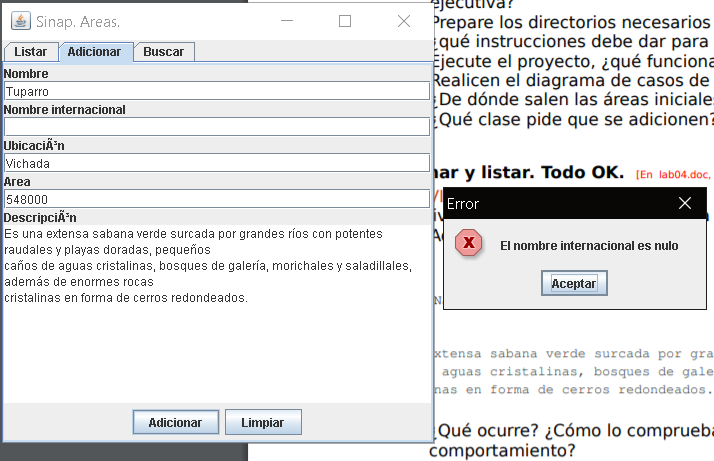
**3. Analicen el diseño realizado. ¿Qué método debería lanzar la excepción? ¿Qué métodos deberían propagarla? ¿Qué método debería atenderla? Explique claramente.**

La excepción es lanzada en el método adiciones de la clase Sinap y propagada en la clase área en el método constructor y atendida en SinapGUI que es donde se envía el mensaje de la excepción en el botón adicionar.

**4. Construya la solución propuesta. Capturen los resultados de las pruebas.**



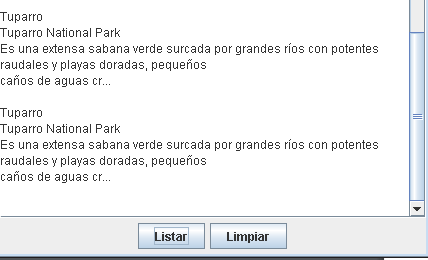
**5. Ejecuten nuevamente la aplicación con el caso de prueba propuesto en 1. ¿Qué sucede ahora? Capture la pantalla.**



**Adicionar un área. ¿Y si ya se encuentra?**

**El objetivo es perfeccionar la funcionalidad de adicionar un área.**

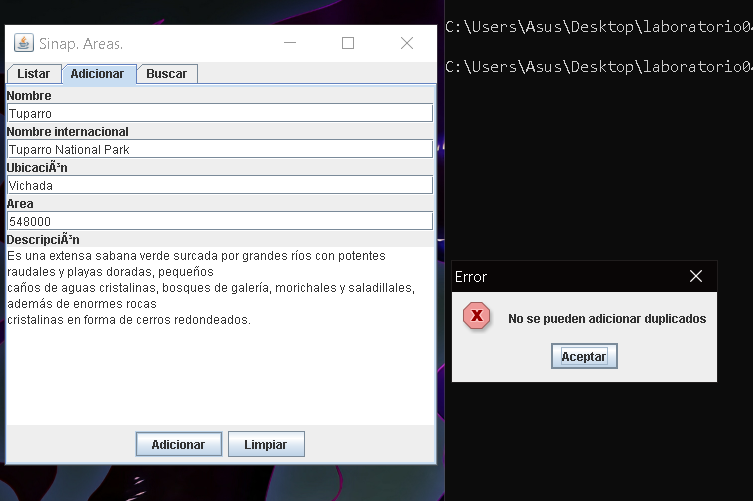
**1. Adicionen dos veces la nueva área Tuparro ¿Qué ocurre? ¿Cómo lo comprueban? Capturen la pantalla. ¿Es adecuado este comportamiento?**

Se añade repetido, mirándolo al listar. No es adecuando el comportamiento pues añade repetidas veces la misma área sin excepción ni control.

**2. Analicen el diseño realizado. ¿Qué método debería lanzar la excepción? ¿Qué métodos deberían propagarla? ¿Qué método debería atenderla? Explique claramente.**

**3. Construya la solución propuesta. Capturen los resultados de las pruebas.**

**4. Ejecuten nuevamente la aplicación con el caso de prueba propuesto en 1., ¿Qué sucede ahora? Capture la pantalla**.



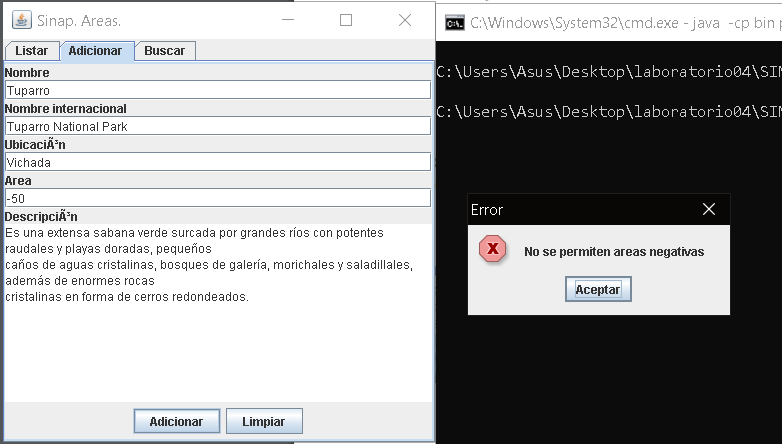
**Adicionar un área. ¿Otras condiciones?**

**El objetivo es perfeccionar la funcionalidad de adicionar un área.**

**1. Propongan nuevas condiciones para que la adición de un área sea más robusta.**

Propusimos una

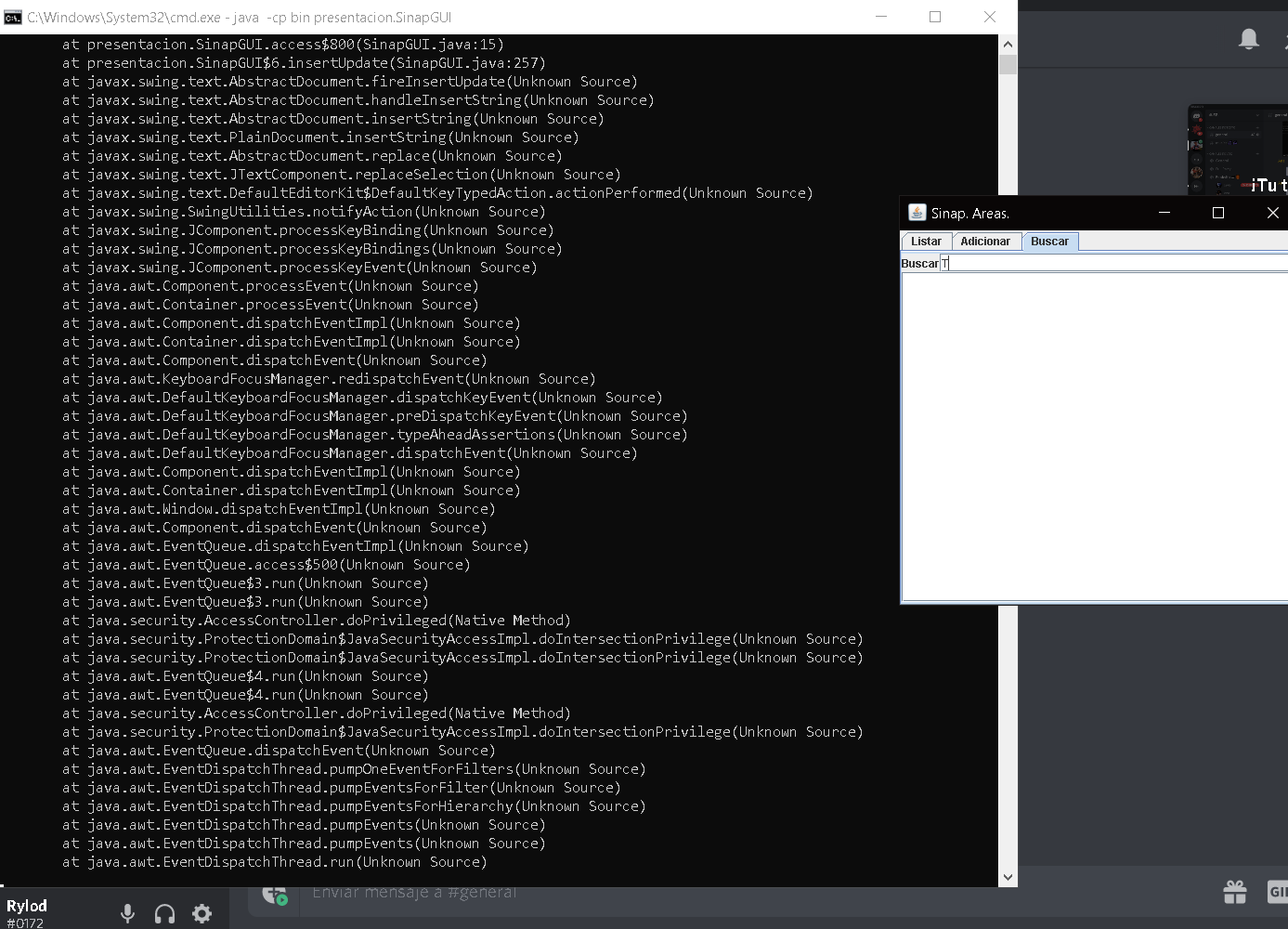
**2. Construya la solución propuesta (diseño, prueba de unidad, código). Capturen los resultados de las pruebas.**



**Consultando por patrones. ¡ No funciona y queda sin funcionar!**

**1. Consulten un área especial que inicie con T. ¿Qué sucede? ¿Qué creen que pasó? Capturen el resultado. ¿Quién debe conocer y quien NO debe conocer esta información?**

No esta aun implementado la función de buscar, ya que arroja un NullPointerException



**2. Exploren el método registre de la clase Registro ¿Qué servicio presta?**

Atrapa las excepciones causadas y las registra

**3. Analicen el punto adecuado para que SIEMPRE, al sufrir en cualquier punto el sistema un incidente como este, se presente un mensaje especial de alerta al usuario, se guarde la información del error en el registro de error y termine la ejecución. Expliquen y construyan la solución.**

Se llama a el método registre en donde se atiende la excepción para registrarla y ahí mismo se usa el JOptionPane para dar un mensaje amigable al usuario de error y terminar la ejecución.

**4. Ejecuten nuevamente la aplicación con el caso propuesto en 1. ¿Qué mensaje salió en pantalla? ¿La aplicación termina? ¿Qué información tiene el archivo de errores?**

Ocurre un NullPointerException, la aplicación no termina. La información del error completo es Exception in thread "AWT-EventQueue-0" java.lang.NullPointerException

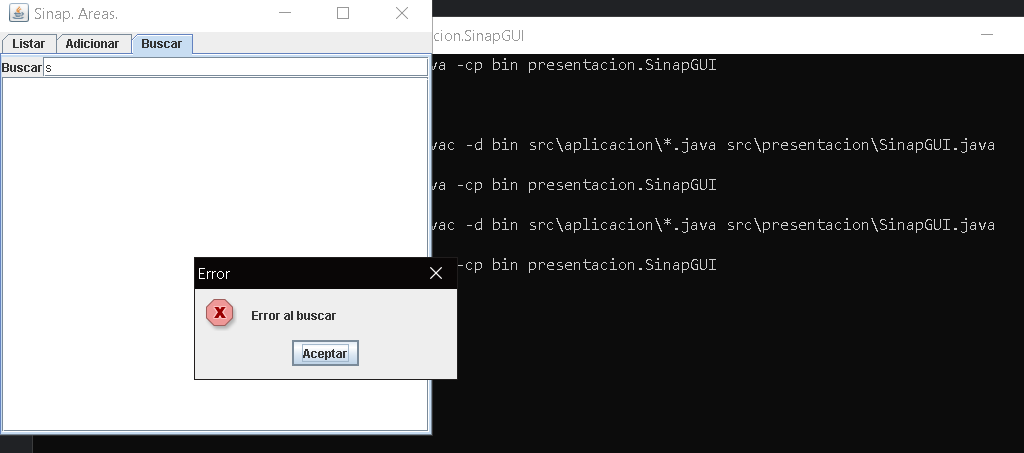
**5. ¿Es adecuado que la aplicación continúe su ejecución después de sufrir un incidente como este? ¿de qué dependería continuar o parar?**

Es adecuado que continúe si el error se refiere a no encontrar en la lista algo con el nombre “T”, en otro caso la ejecución debería detenerse.

**6. Analicen el punto adecuado para que EN ESTE CASO se presente un mensaje especial de alerta al usuario, se guarde la información del error en el registro y continúe la ejecución. Expliquen y construyan la solución. No eliminen la solución de 3.**

Se colocaron JOptionPanel con mensajes de error cuando ocurra un error en la búsqueda, esto en la clase SinapGUI

**7. Ejecuten nuevamente la aplicación con el caso propuesto en 1. ¿Qué mensaje salió en pantalla? ¿La aplicación termina? ¿Qué información tiene el archivo de errores?**

No tiene que terminar la aplicacion cuando ha un error en la busqueda y la informacion guarda la fecha, hora, el nombre completo de la clase y el método y en el caso de mensajes de nivel grave la línea de código donde se generó el reporte, junto al error.

**Consultando por patrones. ¡Ahora si funciona!**

**1. Revisen el código asociado a buscar en la capa de presentación y la capa de aplicación. ¿Qué método es responsable en la capa de presentación? ¿Qué método es responsable en la capa de aplicación?**

accionBuscar es el método responsable de la capa de presentacion

**2. Realicen ingeniería reversa para la capa de aplicación para buscar. Capturen los resultados de las pruebas. Deben fallar.**

**3. ¿Cuál es el error? Soluciónenlo. Capturen los resultados de las pruebas.**

En Sinap lo que se usa para la búsqueda de iguala a null, en lugar de una ArrayList

**4. Ejecuten la aplicación nuevamente con el caso propuesto. ¿Qué tenemos en pantalla? ¿Qué información tiene el archivo de errores?**

La información del archivo de error guarda la fecha, hora, el nombre completo de la clase y el método y en el caso de mensajes de nivel grave la línea de código donde se generó el reporte, junto al error.

**RETROSPECTIVA**

**1. ¿Cuál fue el tiempo total invertido en el laboratorio por cada uno de ustedes?**

**(Horas/Hombre)**

Javier Esteban López 8h

Andres Felipe Davila 8h

**2. ¿Cuál es el estado actual del laboratorio? ¿Por qué?**

**El laboratorio está finalizado falto una prueba**

**3. Considerando las prácticas XP del laboratorio. ¿cuál fue la más útil? ¿por qué?**

La práctica más importante sigue siendo la de la prueba de unida porque haciendo las respectivas pruebas podemos realizar

**4. ¿Cuál consideran fue el mayor logro? ¿Por qué?**

El mayor logro fue lograr compilar las pruebas en consola ya que al comienzo nos pareció bastante confuso

**5. ¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para resolverlo?**

Lograr compilar en consola y configurar bien las variables de entorno para ejecutar todo

**6. ¿Qué hicieron bien como equipo? ¿Qué se comprometen a hacer para mejorar los**

**resultados?**

A pesar de estar en cuarentena utilizamos la app discord para mantenernos comunicados y trabajar en los laboratorios