

# Práctica Scrum



**Realizado por:**  
Danil Loginov  
David Rasero

## - Introducción

Nuestro proyecto se va a basar en una plataforma de contratación para empresas en la que podremos archivar a nuestros trabajadores de una manera eficiente, pudiendo almacenar sus datos personales, cargo en la empresa y salario, asimismo podremos cambiar los datos de los trabajadores en caso necesario.

## - Marcaremos los objetivos a tener.

Crear una interfaz de usuario intuitiva: El objetivo sería diseñar una interfaz fácil de usar y atractiva para que los usuarios puedan utilizar la aplicación de manera eficiente. ✓

Implementar funcionalidades clave: Identificar las funcionalidades principales de la aplicación, como la selección de puestos, el ingreso de datos del trabajador, la gestión de salarios, etc. El objetivo sería desarrollar e implementar estas características esenciales en el producto final. ✓

Establecer opciones para cambios de datos repentinos: Desarrollar una opción que permita modificar los datos que se guardaron en un principio con el fin actualizar la información de cada trabajador. ✓

Optimizar el proceso de búsqueda: Implementar un sistema de búsqueda eficiente que permita a los usuarios encontrar rápidamente los puestos de trabajo disponibles según sus criterios, como el tipo de trabajo, departamento, salario, etc. Y garantizar que no se puedan repetir trabajadores con datos únicos como el dni. ✓

Comentar el código del proyecto para que sea más fácil de seguir: Por cada método o clase que introdujamos, un pequeño texto explicándolo. ✓

Entregar iteraciones incrementales: Siguiendo la filosofía de Scrum, el objetivo es entregar iteraciones incrementales de la aplicación en cada Sprint, lo que significa que se irán agregando funcionalidades gradualmente y el producto se irá mejorando a lo largo del tiempo. ✓

## - Miembros del equipo Scrum

Scrum Master: Es el facilitador y líder del equipo Scrum. Su responsabilidad principal es asegurarse de que se sigan los principios y prácticas de Scrum. El Scrum Master ayuda a eliminar obstáculos, fomenta la comunicación y promueve la colaboración dentro del equipo. En este caso los Scrum Master serían los profesores.

Product Owner: Es el representante de los stakeholders o interesados y es responsable de maximizar el valor del producto. El Product Owner define y prioriza los elementos del backlog del producto, trabaja en estrecha colaboración con el equipo de desarrollo y toma decisiones acerca de qué funcionalidades se implementan en el producto. Se podría decir que formamos parte tanto nosotros como el profesorado ya que, Danil y David decidimos cómo enfocar el proyecto a nuestra manera y los docentes nos guían e indican cómo progresar.

Equipo de Desarrollo: Es el grupo de personas encargadas de convertir los elementos del backlog del producto en incrementos de software funcionales. Formado por Danil y David, los encargados de llevar a cabo el desarrollo del proyecto.

- **Dividimos los sprints necesarios**

<p><b>Primer Sprint (Completado)</b></p>	<p><u>Reunión de Planificación del Sprint:</u> Comenzamos con una lluvia de ideas para plantear el tema sobre el que realizaremos el proyecto. Tras haber descartado varias ideas nos hemos inclinado por hacer una base de datos de trabajadores.</p> <p><u>Reuniones Diarias:</u> En esta reunión hemos determinado que David realizará la clase Trabajador y Departamento y Danil la clase GuardarTrabajador.</p> <p><u>Reunión de Revisión del Sprint:</u> No hemos tenido esta reunión ya que las clases que hemos creado no tenían fallos y se adaptan perfectamente a nuestro proyecto.</p> <p><u>Reunión de Retrospectiva del Sprint:</u> En esta reunión decidimos que el sprint está terminado y procedemos al 2º sprint.</p>
<p><b>Segundo Sprint (Completado)</b></p>	<p><u>Reunión de Planificación del Sprint:</u> En este encuentro digital, decidimos crear las clases de GuardarTrabajador, diseñar la interfaz gráfica e implementar el botón de CargarTrabajador.</p> <p><u>Reuniones Diarias:</u> Planteamos la división de tareas de la siguiente forma: David → Crea la clase GuardarTrabajador Danil → Diseña la interfaz gráfica e implementa el botón de CargarTrabajador.</p> <p><u>Reunión de Revisión del Sprint:</u> La clase de GuardarTrabajador no ha presentado ningún fallo, sin embargo la implementación del botón CargarTrabajador ha dado pela ya que teníamos que pensar una forma óptima de poder almacenar la lista de los trabajadores en la clase GuardarTrabajador para que el botón de CargarTrabajador se visualizara de forma correcta.</p> <p><u>Reunión de Retrospectiva del Sprint:</u> Una vez solucionados los problemas de CargarTrabajador damos por finalizado el 2º Sprint.</p>

<p><b>Tercer Sprint (Completado)</b></p>	<p><u>Reunión de Planificación del Sprint:</u> Progresamos con la interfaz y planteamos el botón de calcularSueldo y clase despedirTrabajador.</p> <p><u>Reuniones Diarias:</u> Repartimos la tarea de la siguiente manera: David → Se encarga del botón calcularSueldo Danil → Se encarga de la clase despedirTrabajador y su botón en la interfaz.</p> <p><u>Reunión de Revisión del Sprint:</u> Nos hemos dado cuenta de que la clase calcularSueldo no funciona ya que no es compatible con la forma de almacenar datos en la clase GuardarTrabajador así que ayudándonos de la clase despedirTrabajador solucionamos el problema arreglando el código.</p> <p><u>Reunión de Retrospectiva del Sprint:</u> Viendo la correcta funcionalidad de la clase y de ambos botones damos por finalizado este sprint.</p>
<p><b>Cuarto Sprint (Completado)</b></p>	<p><u>Reunión de Planificación del Sprint:</u> Seguimos con la interfaz gráfica, en este caso con las funcionalidades CambiarNombre, CambiarTeléfono y CambiarDíasTrabajados.</p> <p><u>Reuniones Diarias:</u> Repartimos la tarea de la siguiente forma : David → Se encarga del botón CambiarNombre Danil → Se encarga de los apartados CambiarTeléfono y CambiarDíasTrabajados.</p> <p><u>Reunión de Revisión del Sprint:</u> El apartado de CambiarNombre tendría más dificultad con respecto al resto de apartados porque tendríamos que cambiar dos ficheros a la vez con lo cual concluimos resolviendo este apartado entre los dos.</p> <p><u>Reunión de Retrospectiva del Sprint:</u> Después de solucionar los problemas, dimos por completado este sprint.</p>

<p><b>Quinto Sprint (Completado)</b></p>	<p><u>Reunión de Planificación del Sprint:</u> Seguimos con la interfaz gráfica, implementamos CambiarContrato, CambiarDepartamento y CambiarPuesto.</p> <p><u>Reuniones Diarias:</u> Repartimos la tarea de la siguiente forma : David → Se encarga del botón CambiarContrato Danil → Se encarga de los apartados CambiarDepartamento y CambiarPuesto. Estos tres los hicimos conjuntamente ya que el código era muy similar.</p> <p><u>Reunión de Revisión del Sprint:</u> Al ser tan parecidos estos botones y hacerlos conjuntamente no supuso gran problema.</p> <p><u>Reunión de Retrospectiva del Sprint:</u> Dimos por completado este sprint tras finalizar los botones.</p>
<p><b>Sprint Final</b></p>	<p><u>Reunión de Planificación del Sprint:</u> Ya con el trabajo casi hecho, nos ponemos manos a la obra para comprobar su eficacia.</p> <p><u>Reuniones Diarias:</u> Entre ambos vamos recopilando los diferentes errores que aparecían y los corregimos.</p> <p><u>Reunión de Revisión del Sprint:</u> Realmente apenas tuvimos errores al final del programa ya que los que iban surgiendo los corregimos al instante.</p> <p><u>Reunión de Retrospectiva del Sprint:</u> Finalmente damos por completado este último sprint y el proyecto.</p>

#### - Ceremonias scrum del proyecto.

Reunión de Planificación del Sprint: Esta ceremonia se realiza al comienzo de cada Sprint para planificar el trabajo que se realizará durante ese período. Además, se establecen los objetivos del Sprint y se desglosa el trabajo en tareas más pequeñas y estimables. Cada vez que comenzamos un sprint nos conectamos por llamada y decidimos que vamos a añadir a nuestro proyecto siempre con la retrospectiva del compañero.

Reuniones Diarias: Estas breves reuniones se llevan a cabo diariamente durante el Sprint y suelen durar entre 15 y 20 minutos (vía online). Nos reunimos para sincronizar el trabajo y compartir el progreso desde la última reunión.

Reunión de Revisión del Sprint: Al final de cada Sprint, se lleva a cabo la reunión de revisión del Sprint. Se muestra el incremento del proyecto y se recopilan comentarios y sugerencias.

Escuchamos las críticas constructivas y retocamos el trabajo para que quede lo más limpio y ordenado posible.

Reunión de Retrospectiva del Sprint: Inmediatamente después de la reunión de revisión del Sprint, se lleva a cabo la reunión de retrospectiva del Sprint. En esta ceremonia, el equipo de Scrum reflexiona sobre el último Sprint y analiza cómo fue el proceso y cómo se pueden hacer mejoras. Damos por finalizado el sprint y proseguimos con el trabajo.

#### - **Ventajas o manifiestos Scrum**

Hemos notado que los Sprints permiten iterar rápidamente y ajustar el enfoque según sea necesario. Esto facilita la toma de decisiones y la comunicación efectiva, permite obtener retroalimentación temprana, lo que a su vez ayuda a mejorar el producto a lo largo del tiempo.

La estructura de roles claros y las ceremonias regulares fomentan una comunicación efectiva, el intercambio de conocimientos y la toma de decisiones conjunta.

Scrum promueve la interacción diaria en las reuniones diarias para maximizar la comunicación. En Scrum, se prioriza el desarrollo de incrementos de producto en cada Sprint. Los Sprints permiten una mayor flexibilidad y capacidad de ajuste a medida que se obtienen nuevos conocimientos o se producen cambios en los requisitos.

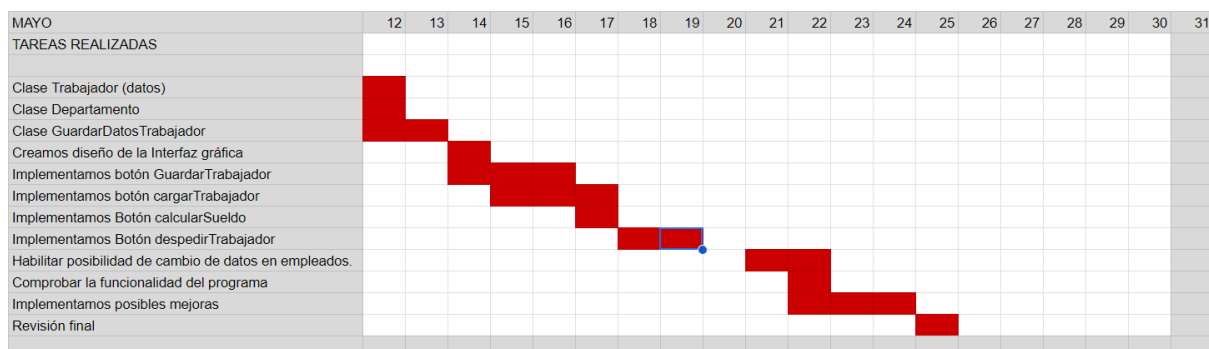
Estas ventajas y principios son fundamentales en Scrum y reflejan la filosofía ágil de desarrollo de software. Scrum proporciona un enfoque estructurado y adaptable para la gestión de proyectos, lo que permite un mayor enfoque en la entrega de valor.

#### - **Trello de planificación (compartido)**

<https://trello.com/b/woa0SEbi/scrum>

#### - **Diagrama de gantt**

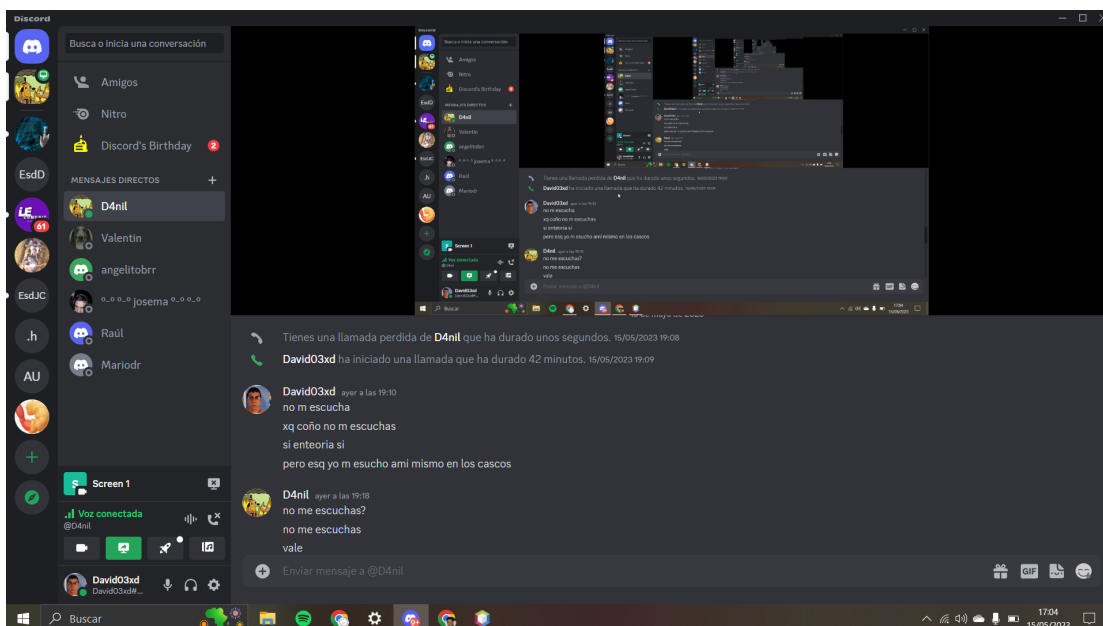
Realizado con excel.



- **Aplicaciones que nos ayuden a utilizar la metodología scrum.**

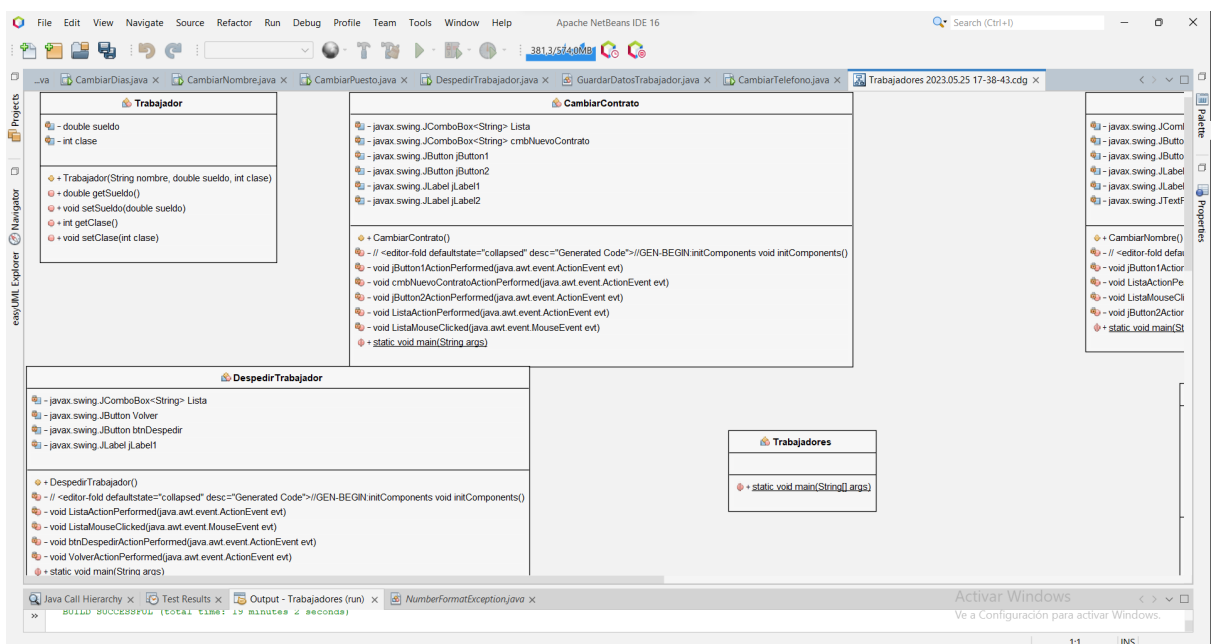
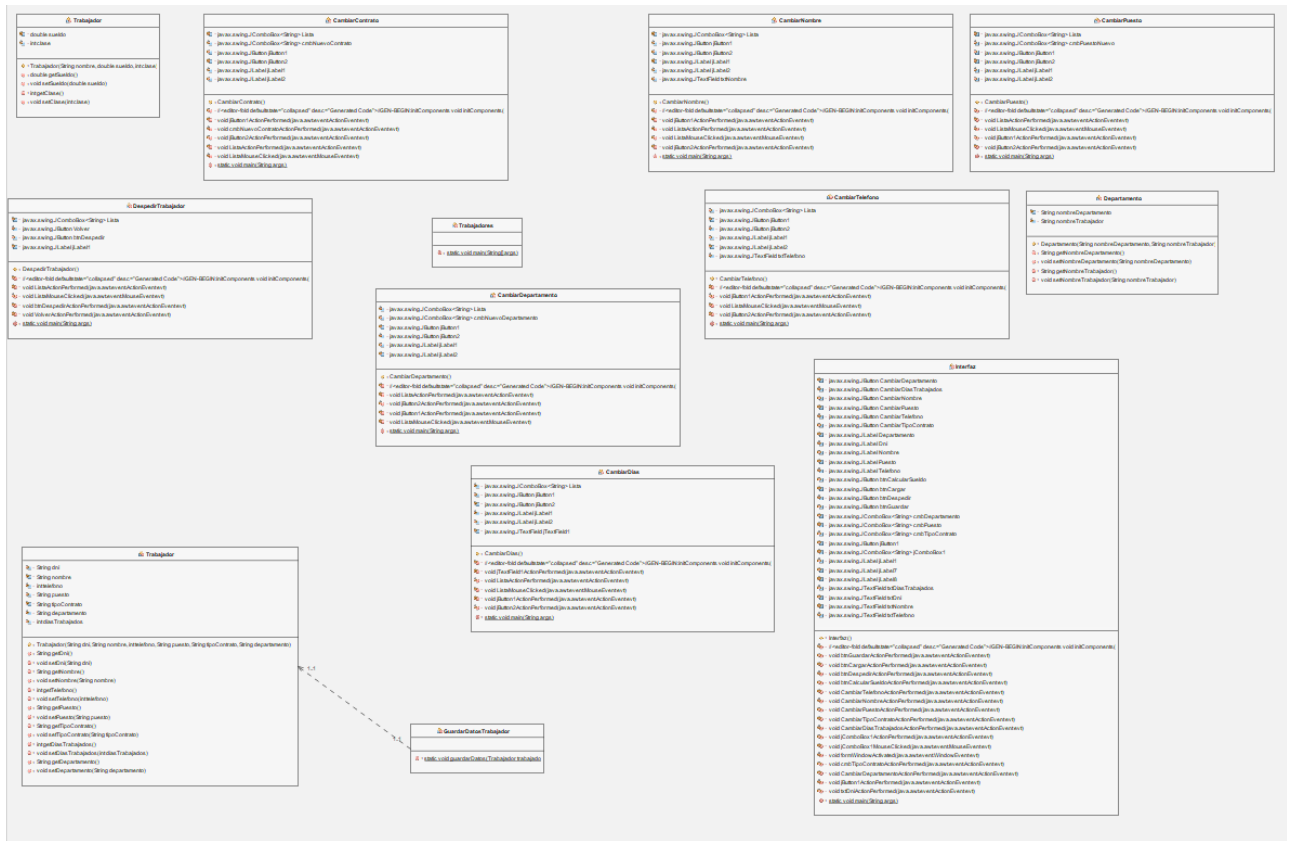
Existen varias aplicaciones y herramientas que pueden ayudarte a implementar y gestionar la metodología Scrum en tus proyectos. Proporciona funcionalidades completas para planificar, rastrear y gestionar proyectos Scrum, por ejemplo Trello, Scrumwise, Azure Devops o Jira Software. Ofrece características específicas para Scrum, como la creación de Product Backlog, la planificación de Sprints, la gestión de tareas y la generación de informes de rendimiento. Además, hay complementos y plantillas disponibles para agregar características de Scrum a tus tableros.

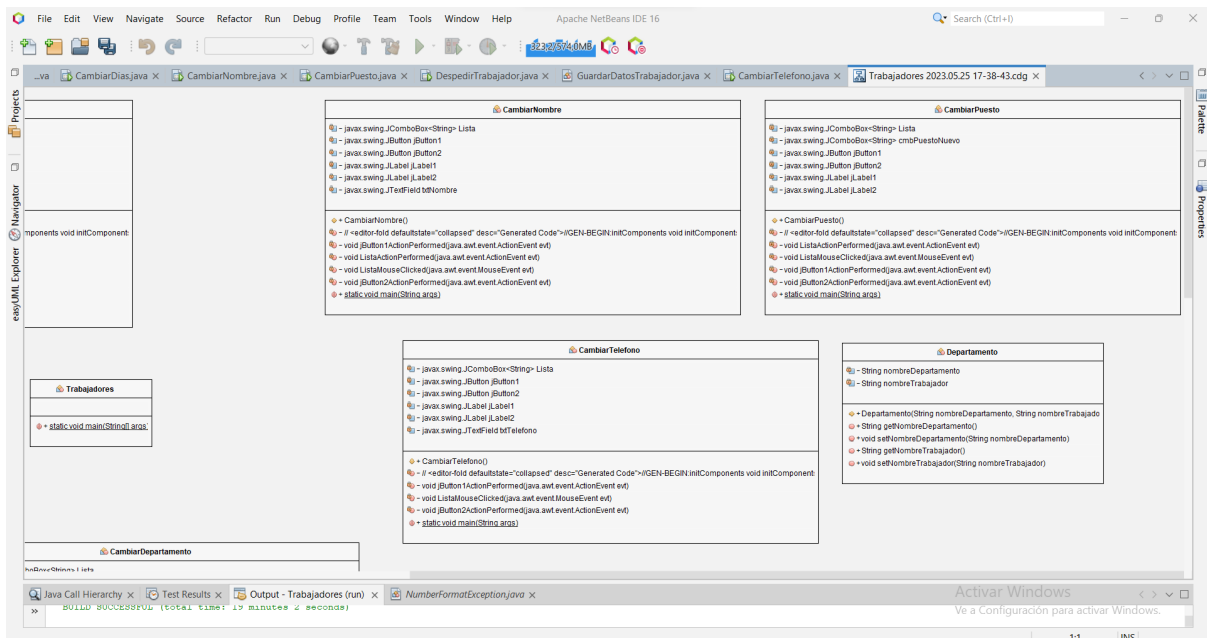
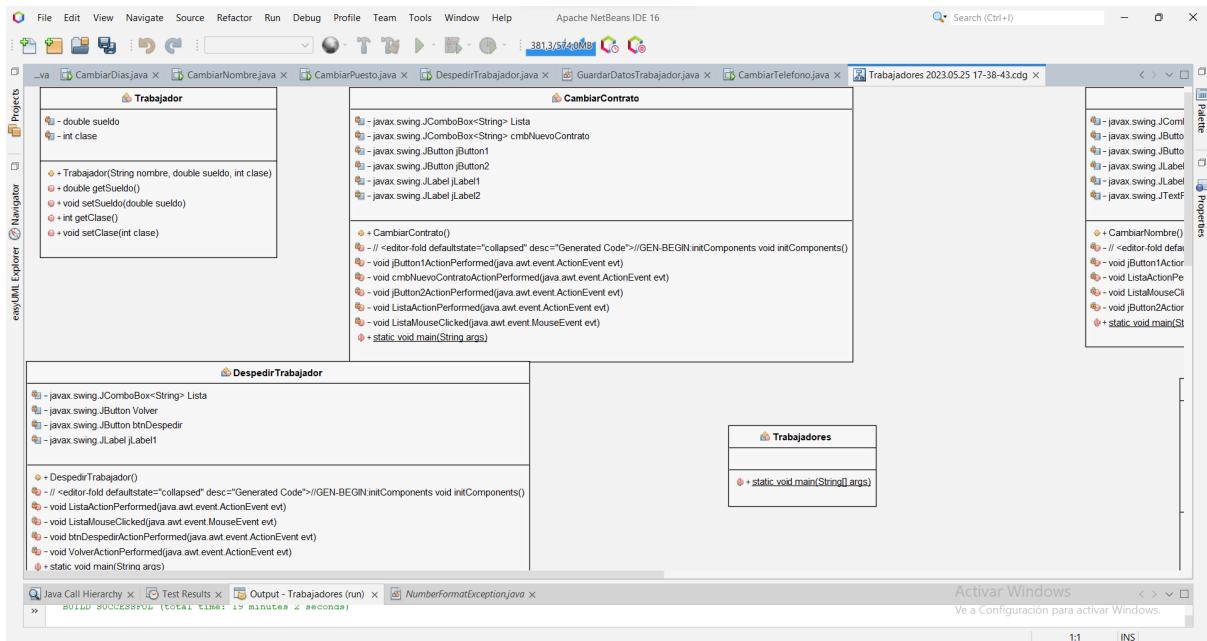
Com es una plataforma de gestión de proyectos y colaboración que puede adaptarse a la metodología Scrum. En nuestro caso hemos seleccionado Trello ya que sus funcionalidades son intuitivas y fáciles de manejar, para la comunicación simplemente Whatsapp para un chat rápido y Discord para llamadas y compartir pantalla.

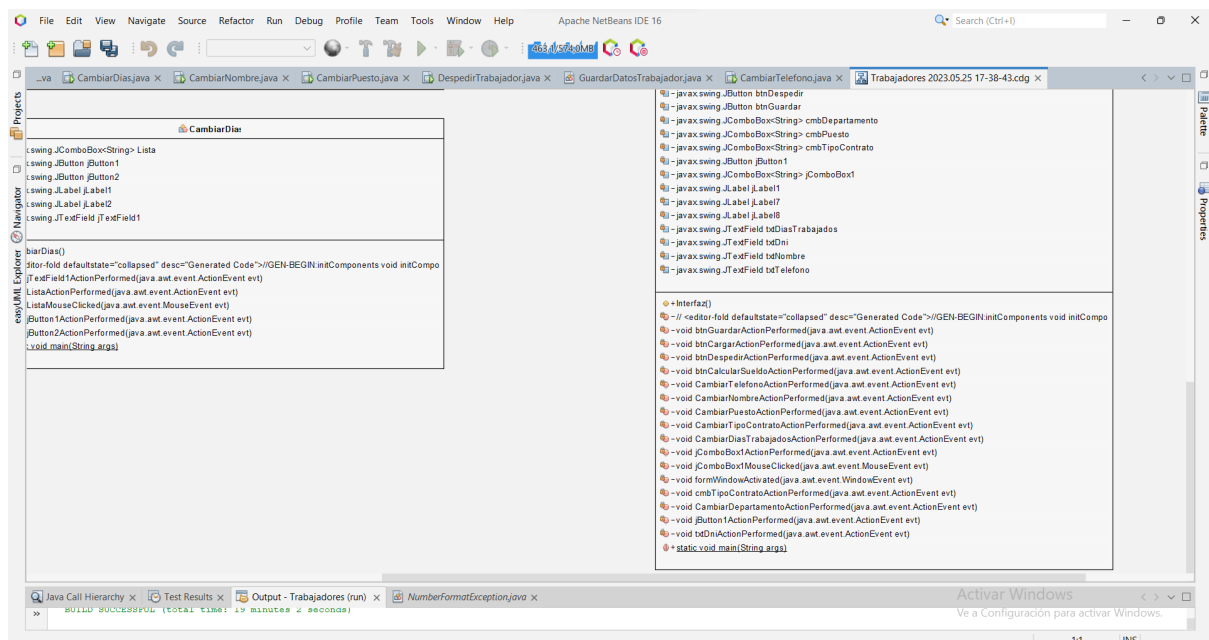




- **Diagrama de clases de los objetos del proyecto**







- Diagrama de casos de uso

