



Manual de Desarrollo

Versión 0.1

**Ensenada, Baja
California**

**8 de Noviembre de
2023**



Índice

Introducción	3
Definiciones	3
Roles	4
HU2.1 - Pantalla de inicio	5
HU2.2 - Edición de solicitud de salida	6
HU2.3 - Cancelar solicitud de salida	9
HU2.4 - Crear solicitud de salida	11
HU2.5 - Listar solicitudes de salida para revisión	14
HU2.6 - Rechazo de solicitud de salida	16
HU2.7 - Firmado de solicitud de salida	18
HU2.8 - Subir reporte	20
HU2.9 - Consultar solicitud de salida	22



Historias de Usuario Módulo 2

Introducción

El presente documento pretende servir como una guía para todos aquellos que necesiten instalar, trabajar en, y hacer cambios al código fuente del sistema de gestión de salidas.

Este manual guía por el proceso de instalar las dependencias, clonar el repositorio git con el código y configurar un entorno de desarrollo integrado (IDE) para poder hacer cambios y contribuir al desarrollo y mantenimiento. Además, presenta unos lineamientos que se usan en el repositorio para mantener la consistencia.

Dependencias

Las dependencias que se deben tener instaladas en el sistema son las siguientes.

- [Java JDK 17](#).
- [Apache Tomcat 9](#).
- Una herramienta de git, ya sea [git en la terminal](#) o un cliente gráfico como [GitHub Desktop](#). En realidad no es estrictamente necesario, ya que muchos editores de texto y IDE tienen integradas funcionalidades de git, (IntelliJ IDEA siendo uno de esos), pero tener una instalación separada de git es más práctico por lo que en este manual se va a asumir que se usa una instalación propia de git.
- Un editor de texto o IDE. Cualquiera podría ser usado, pero recomendamos [IntelliJ IDEA](#).

Herramientas adicionales que podrían ser útiles, pero que no son esenciales para construir y ejecutar el proyecto.

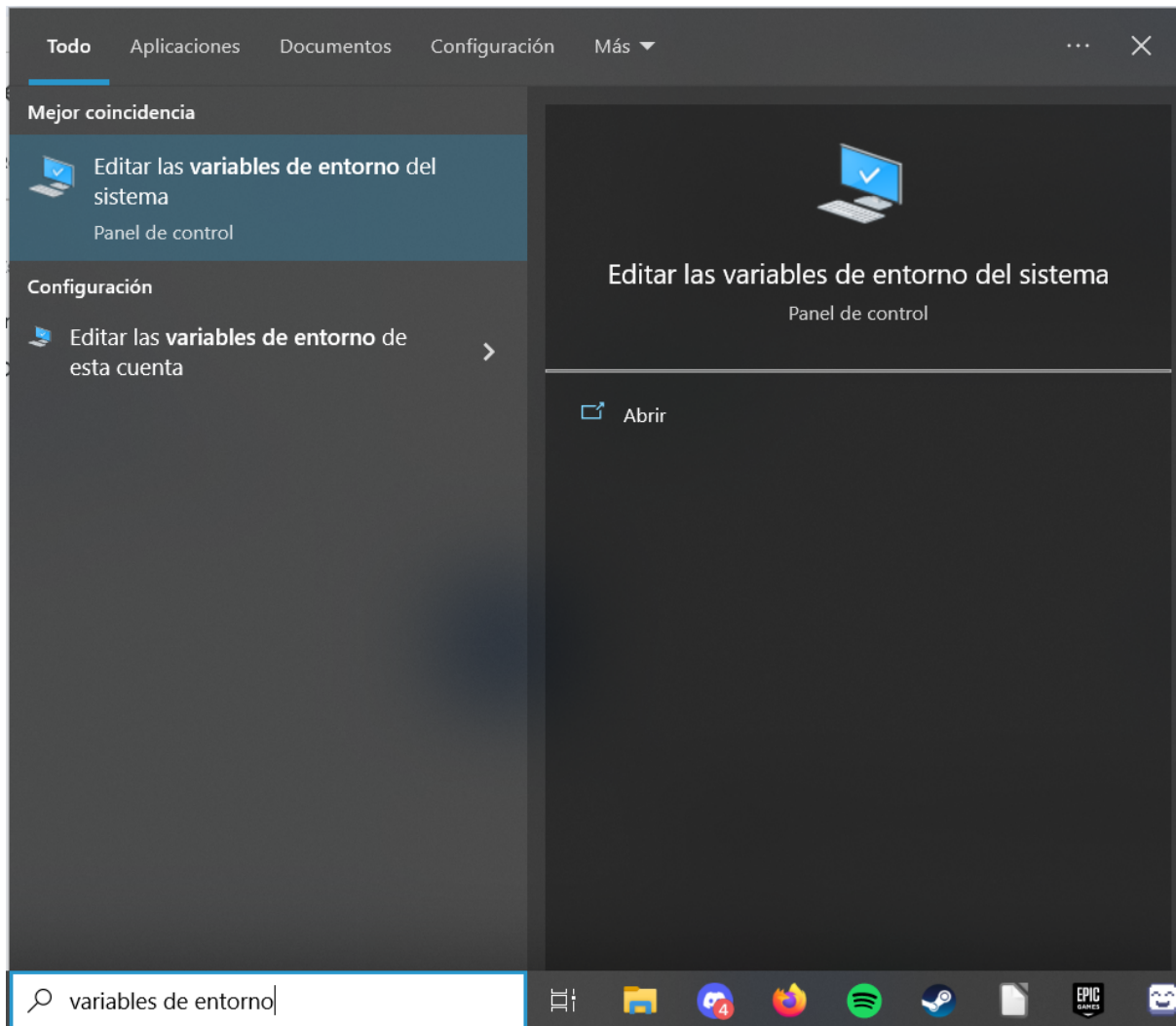
- [Maven 3.9.x](#). En caso de que prefiera usar una versión propia de maven, ya que el mismo repositorio incluye un wrapper para poder usarlo.

Instalar Java 17

1. Primero descargar el JDK desde la [página de descarga oficial](#), descargar la versión correspondiente para el sistema.
2. Ejecutar el instalador y seguir las instrucciones indicadas.



3. En el menú de búsqueda de windows, introducir “variables de entorno” y presionar enter.



4. En la ventana abierta, seleccionar la opción “variables de entorno”.



Propiedades del sistema



Nombre del equipo	Hardware
Opciones avanzadas	Protección del sistema
Remoto	

Para realizar la mayoría de estos cambios, inicie sesión como administrador.

Rendimiento

Efectos visuales, programación del procesador, uso de memoria y memoria virtual

Configuración...

Perfiles de usuario

Configuración del escritorio correspondiente al inicio de sesión

Configuración...

Inicio y recuperación

Inicio del sistema, errores del sistema e información de depuración

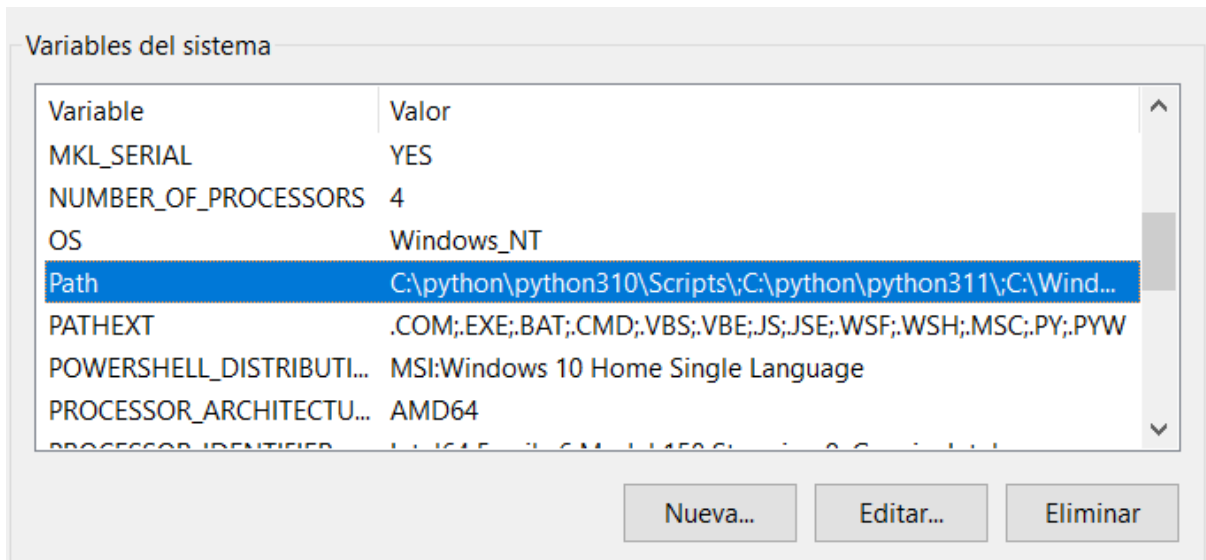
Configuración...

Variables de entorno...

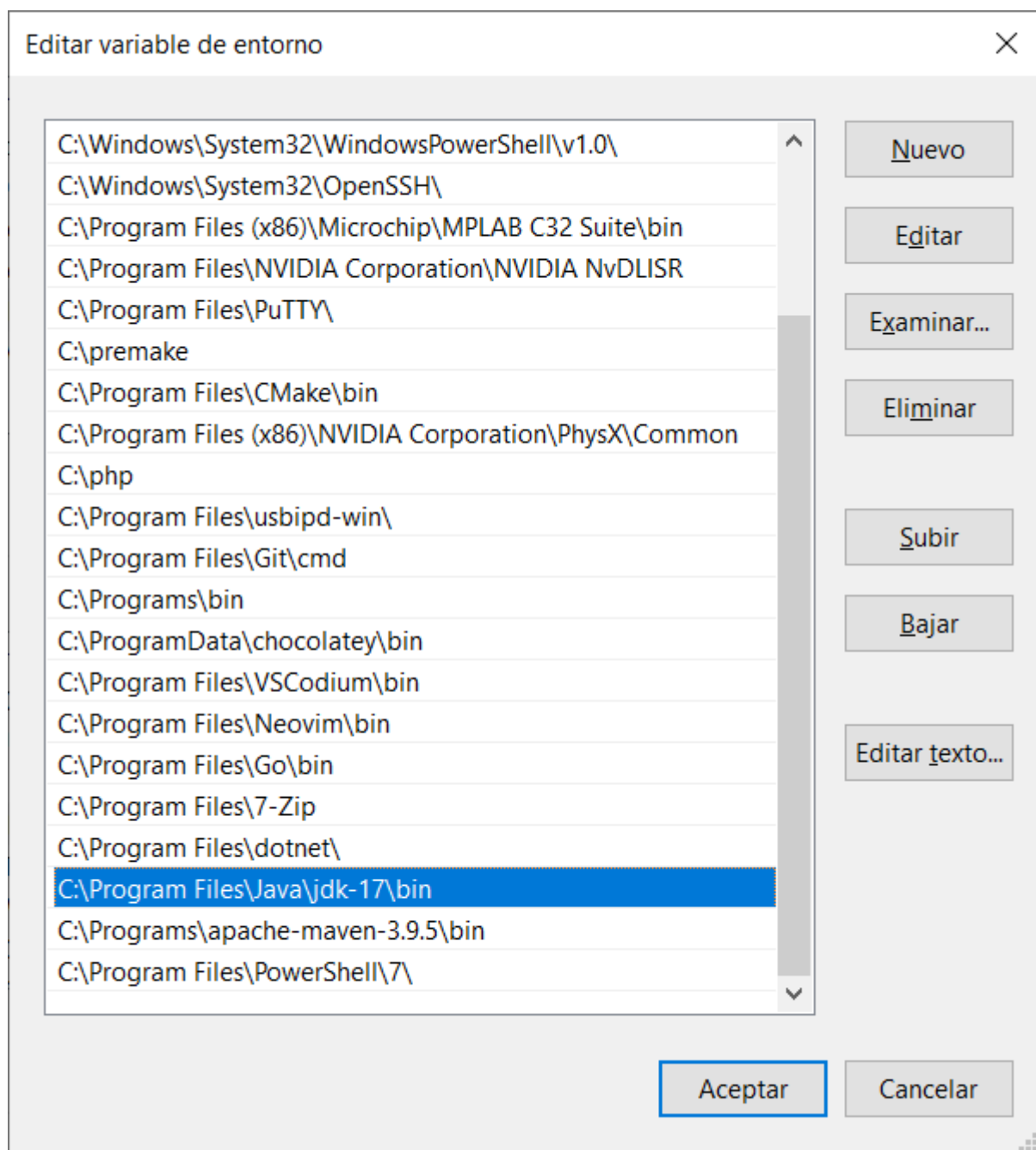
Aceptar Cancelar Aplicar

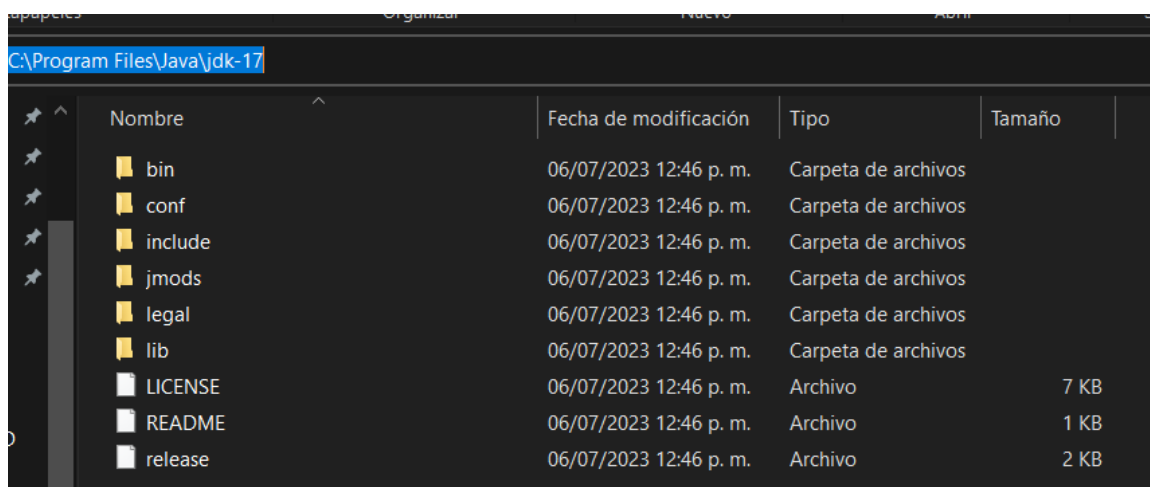
Variables de entorno...

5. Ya sea en la sección de variables de usuario o en la del sistema, se hace doble clic en la variable Path. La diferencia es que si se actualiza la variable del sistema será accesible para todos los usuarios, pero si es de usuario solamente para el actual.

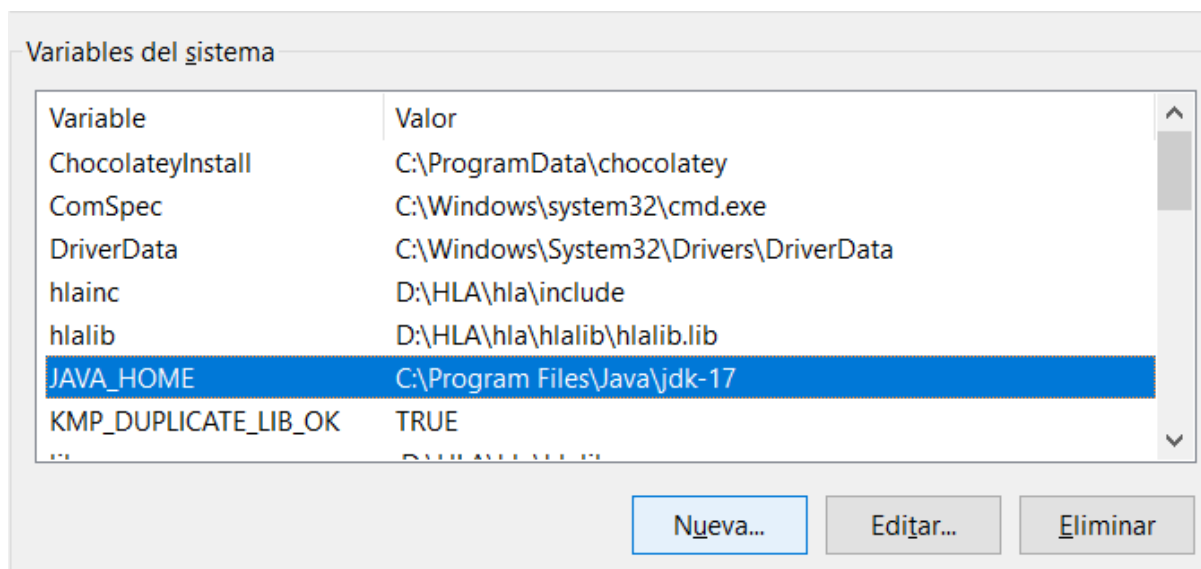


6. Presionar “Nuevo” en la esquina superior derecha y escribir la dirección del directorio donde se instaló el JDK (típicamente C:\Program Files\Java\jdk-<versión>, si no está seguro navegar con el explorador de archivos).

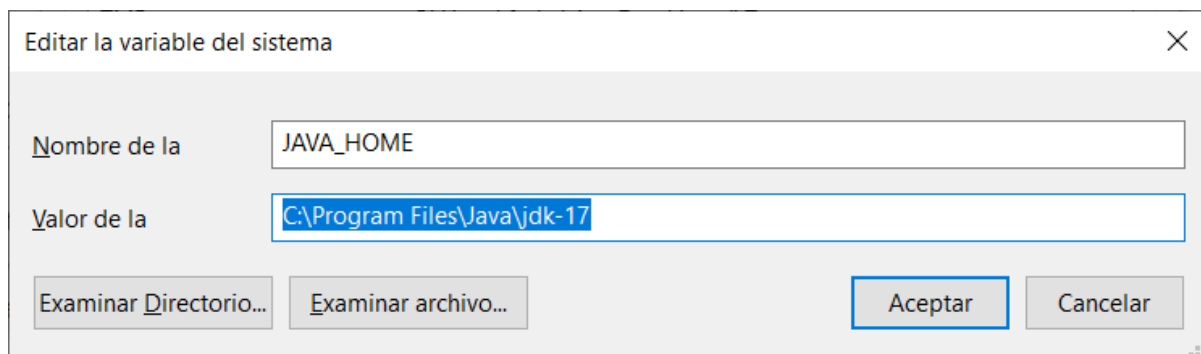




7. Guardar los cambios y regresar a la pantalla anterior.
8. Presionar “Nueva” debajo de la misma sección donde se encuentra la variable Path que acaba de editar.



9. Introducir el nombre como “JAVA_HOME”, el valor siendo la misma dirección que se agregó a Path.



10. Reiniciar el sistema para que los cambios tomen efecto.

Instalar Apache Tomcat

1. En [la página oficial](#), navegar a la sección de distribuciones binarias de la actualización más reciente de la versión 9.0.x.

9.0.82

Please see the [README](#) file for packaging information. It explains what every distribution contains.

Binary Distributions

- Core:
 - [zip](#) ([pgp](#), [sha512](#))
 - [tar.gz](#) ([pgp](#), [sha512](#))
 - [32-bit Windows zip](#) ([pgp](#), [sha512](#))
 - [64-bit Windows zip](#) ([pgp](#), [sha512](#))
 - [32-bit/64-bit Windows Service Installer](#) ([pgp](#), [sha512](#))
- Full documentation:
 - [tar.gz](#) ([pgp](#), [sha512](#))
- Deployer:
 - [zip](#) ([pgp](#), [sha512](#))
 - [tar.gz](#) ([pgp](#), [sha512](#))
- Embedded:
 - [tar.gz](#) ([pgp](#), [sha512](#))
 - [zip](#) ([pgp](#), [sha512](#))

Se recomienda descargar el instalador para windows, que es la última opción dentro de la sección “Core”.



- Core:

- [zip](#) ([pgp](#), [sha512](#))
- [tar.gz](#) ([pgp](#), [sha512](#))
- [32-bit Windows zip](#) ([pgp](#), [sha512](#))
- [64-bit Windows zip](#) ([pgp](#), [sha512](#))
- [32-bit/64-bit Windows Service Installer](#) ([pgp](#), [sha512](#))

Es posible usar los archivos portables zip, pero eso requeriría descomprimirlos en la raíz y configurar el servicio. Si decide tomar esa ruta está bajo su propio riesgo ya que este manual asume que se instaló el servicio mediante el instalador.

2. Ejecutar el instalador y seguir los pasos indicados, dejando las opciones en sus valores predeterminados, hasta que pida configurar un usuario y contraseña.
3. Al momento de introducir un usuario y contraseña se recomienda usar una combinación que sea fácil de recordar, ya que cambiarlas en el futuro es una tarea muy complicada que está fuera del alcance de este manual.
4. Después de configurar el usuario, se introduce el camino al jdk instalado anteriormente, que por defecto se encuentra en en C:\Program Files\Java\jdk-<versión>, simplemente se necesita seleccionar la carpeta donde se encuentra el jdk, por ejemplo jdk-17.
5. Luego de eso, el resto de campos se pueden dejar en sus valores predeterminados.
6. Una vez esté instalado se puede comprobar su funcionamiento entrando a un navegador e introduciendo la url <http://localhost:8080/>.
7. Si el navegador no se puede conectar quiere decir que el servicio no se está ejecutando, por el otro lado, si se presenta alguna página, el servicio fue instalado con éxito.

Git

Se descarga git de la [página oficial](#)

Al igual que Java, se puede seguir la instalación con todos los valores predeterminados. Son muchos valores, y es recomendable leerlos e investigar sobre su significado en la medida de lo posible, pero es posible no cambiar nada y la instalación será perfectamente usable.

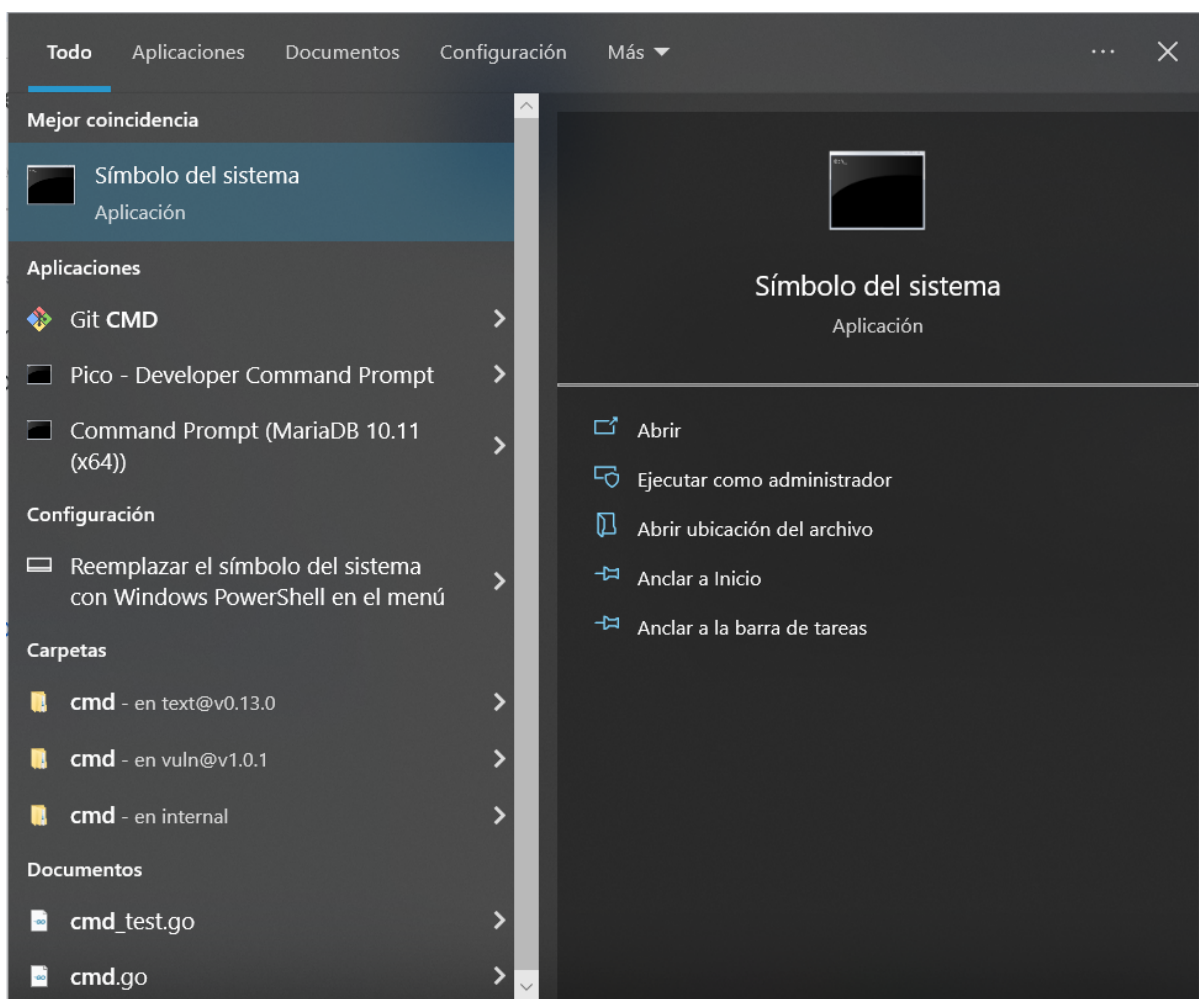


Instalar IntelliJ IDEA

Se instala IntelliJ IDEA desde la [página oficial](#), siguiendo los pasos indicados. Tanto la versión de paga como la gratuita deberían funcionar, aunque en este manual se usa la versión de paga.

Descargar y construir el repositorio

1. Solicitar acceso al repositorio.
2. Clonar el repositorio con su herramienta de preferencia. A continuación se muestran los pasos para usar git desde la terminal.
 - 2.1. En la búsqueda de windows, escribir cmd y seleccionar el símbolo del sistema.





- 2.2. Navegar con `cd` a un directorio donde desea clonar el repositorio.

```
C:\Users\ >cd git\sistema
```

- 2.3. Obtener la url del repositorio.

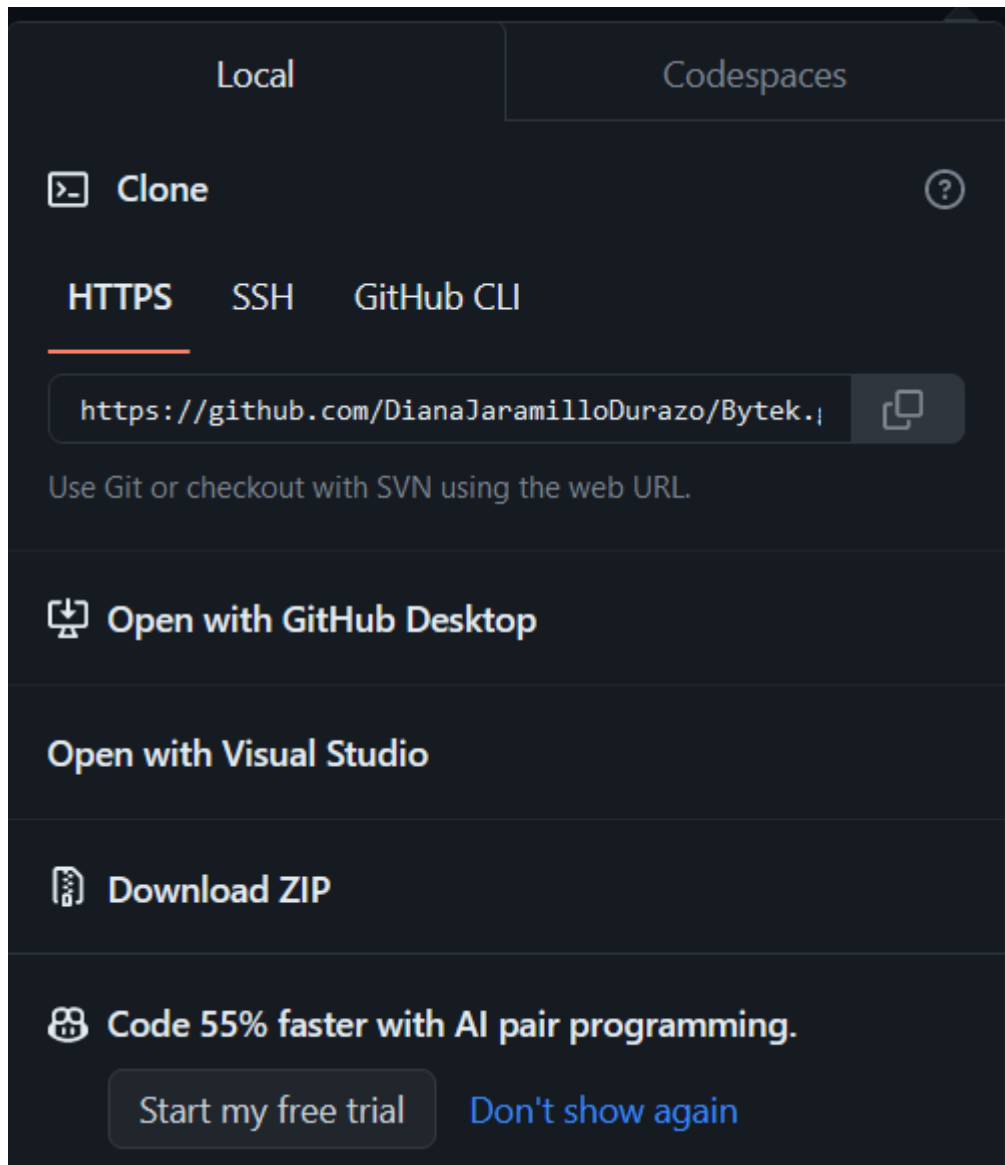
Dentro del menú "Code".

Go to file

Add file ▼

<> Code ▼

Elegir la opción HTTPS y presionar el botón de la derecha para copiarla.



- 2.4. Ejecutar el comando "git clone <url> ." (El "." es importante para no crear una carpeta nueva y en su lugar usar la carpeta actual, si prefiere que git cree una nueva carpeta puede omitir el ".")



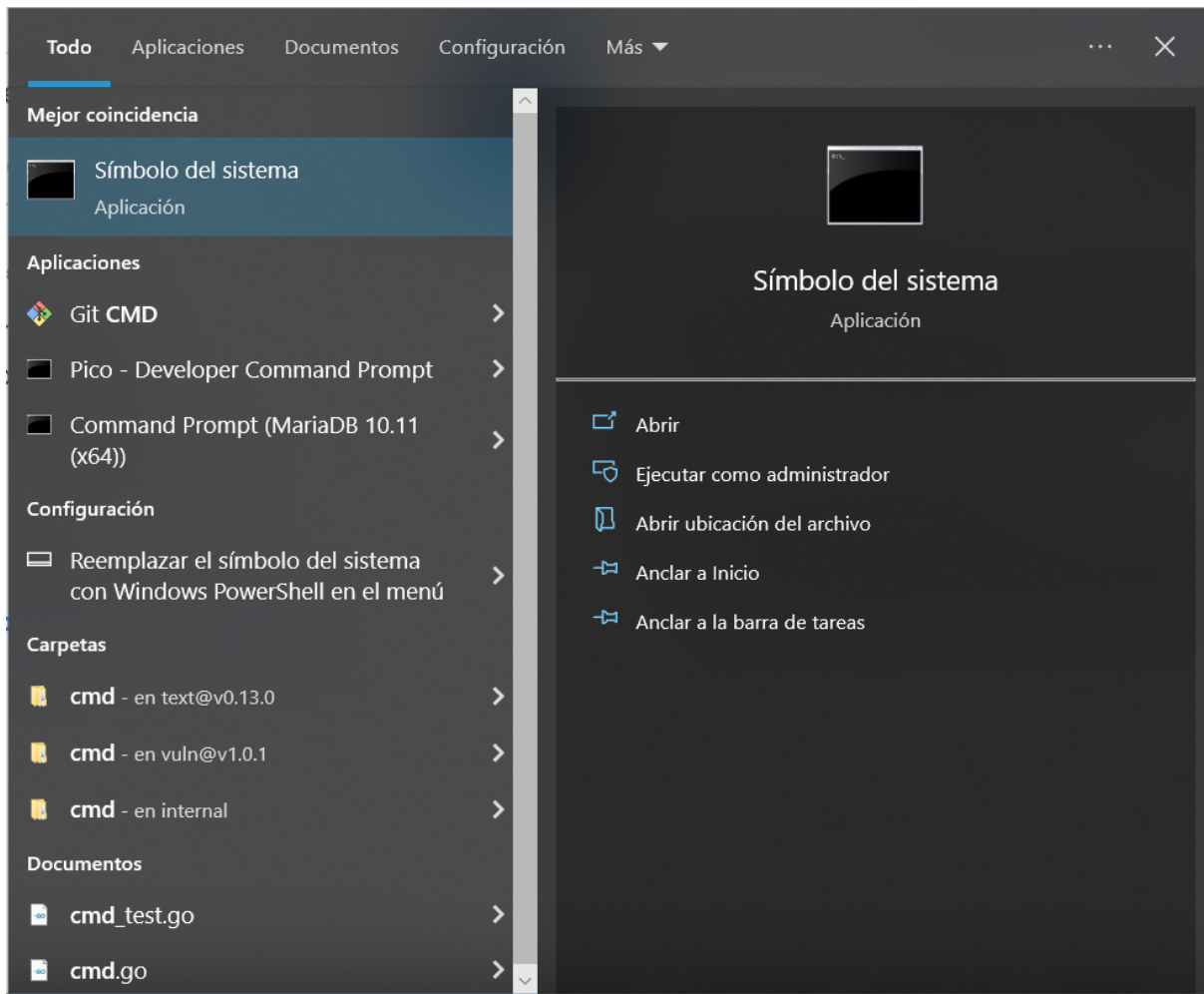
```
C:\Users\... \git\sistema>git clone git@github.com:DianaJaramilloDurazo/Bytek.git .
Cloning into '.'...
remote: Enumerating objects: 796, done.
remote: Counting objects: 100% (796/796), done.
remote: Compressing objects: 100% (388/388), done.
remote: Total 796 (delta 326), reused 704 (delta 237), pack-reused 0
Receiving objects: 86% (685/796), 1.45 MiB | 2.89 MiB/s
Receiving objects: 100% (796/796), 1.46 MiB | 2.89 MiB/s, done.
Resolving deltas: 100% (326/326), done.

C:\Users\... \git\sistema>
```

- 2.5. En el sistema usamos ramas para mantener ciertos aspectos del desarrollo separados, ejecuta el comando “git checkout dev” para cambiar la rama a la de desarrollo.

```
C:\Users\... \git\sistema>git checkout dev
Switched to a new branch 'dev'
branch 'dev' set up to track 'origin/dev'.
```

3. Una vez se tenga clonado el repositorio, se puede probar desde la terminal que todo esté funcionando correctamente. Este paso es opcional, pero puede ayudar a asegurarse de que todo funcione bien antes de proceder a configurar IntelliJ IDEA.
- 3.1. Abrir el símbolo del sistema desde el menú de inicio.



3.2. Navegar con `cd` a la dirección del repositorio.

```
C:\Users\[user]>cd git\sistema
```

3.3. Ejecutar el comando “`mvnw clean compile`”. Si todo sale bien debería mostrar algo similar a lo siguiente.



Si no es el caso, póngase en contacto con soporte.

```

=====
:: Spring Boot ::                (v3.1.4)

2023-11-08T20:44:07.294-08:00 INFO 25512 --- [ restartedMain] com.ubac.fiad.sgs.SgsApplication : Starting SgsApplication using Java 17.0.7 with PID 25512 (C:\Users\ [REDACTED] \git\sistema\target\classes started by in C:\Users\ [REDACTED] \git\sistema)
2023-11-08T20:44:07.298-08:00 INFO 25512 --- [ restartedMain] com.ubac.fiad.sgs.SgsApplication : No active profile set, falling back to 1 default profile: "default"
2023-11-08T20:44:07.370-08:00 INFO 25512 --- [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : Devtools property defaults active! Set 'spring.devtools.add-properties' to 'false' to disable
2023-11-08T20:44:07.372-08:00 INFO 25512 --- [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : For additional web related logging consider setting the 'logging.level.web' property to 'DEBUG'
2023-11-08T20:44:08.533-08:00 INFO 25512 --- [ restartedMain] trationDelegate$BeanPostProcessorChecker : Bean 'org.springframework.ws.config.annotation.DelegatingWsConfiguration' of type 'org.springframework.ws.config.annotation.DelegatingWsConfiguration$$SpringCGLIB$$0' is not eligible for getting processed by all BeanPostProcessors (for example: not eligible for auto-proxying)
2023-11-08T20:44:08.596-08:00 INFO 25512 --- [ restartedMain] .w.s.a.s.AnnotationActionEndpointMapping : Supporting [WS-Addressing August 2004, WS-Addressing 1.0]
2023-11-08T20:44:09.271-08:00 INFO 25512 --- [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8090 (http)
2023-11-08T20:44:09.286-08:00 INFO 25512 --- [ restartedMain] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2023-11-08T20:44:09.287-08:00 INFO 25512 --- [ restartedMain] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.13]
2023-11-08T20:44:09.382-08:00 INFO 25512 --- [ restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2023-11-08T20:44:09.384-08:00 INFO 25512 --- [ restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: Initialization completed in 2009 ms
2023-11-08T20:44:09.732-08:00 INFO 25512 --- [ restartedMain] o.s.b.a.w.s.WelcomePageHandlerMapping : Adding welcome page template: index
2023-11-08T20:44:09.944-08:00 INFO 25512 --- [ restartedMain] o.s.s.web.DefaultSecurityFilterChain : Will secure any request with [org.springframework.security.web.session.DisableEncodeUrlFilter@6a8f720c, org.springframework.security.web.context.request.async.WebAsyncManagerIntegrationFilter@188e31c3, org.springframework.security.web.context.SecurityContextHolderFilter@4a9ce6b8, org.springframework.security.web.header.HeaderWriterFilter@302785fc, org.springframework.security.web.authentication.logout.LogoutFilter@2f74c4c2, org.springframework.security.web.authentication.UsernamePasswordAuthenticationFilter@4e477f91, org.springframework.security.web.savedrequest.RequestCacheAwareFilter@7962b233, org.springframework.security.web.servletapi.SecurityContextHolderAwareRequestFilter@3e95e14a, org.springframework.security.web.authentication.AnonymousAuthenticationFilter@6dc06eb4, org.springframework.security.web.access.ExceptionTranslationFilter@82252b09c, org.springframework.security.web.access.intercept.AuthorizationFilter@8e4fb4]
2023-11-08T20:44:10.308-08:00 INFO 25512 --- [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : LiveReload server is running on port 35729
2023-11-08T20:44:10.348-08:00 INFO 25512 --- [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8090 (http) with context path ''
2023-11-08T20:44:10.360-08:00 INFO 25512 --- [ restartedMain] com.ubac.fiad.sgs.SgsApplication : Started SgsApplication in 3.505 seconds (process running for 4.055)

```





- localhost:8090/login

FACULTAD DE INGENIERÍA ARQUITECTURA Y DISEÑO

SISTEMA DE AUTOMATIZACIÓN PARA SOLICITUDES DE SALIDA

Dirección de correo

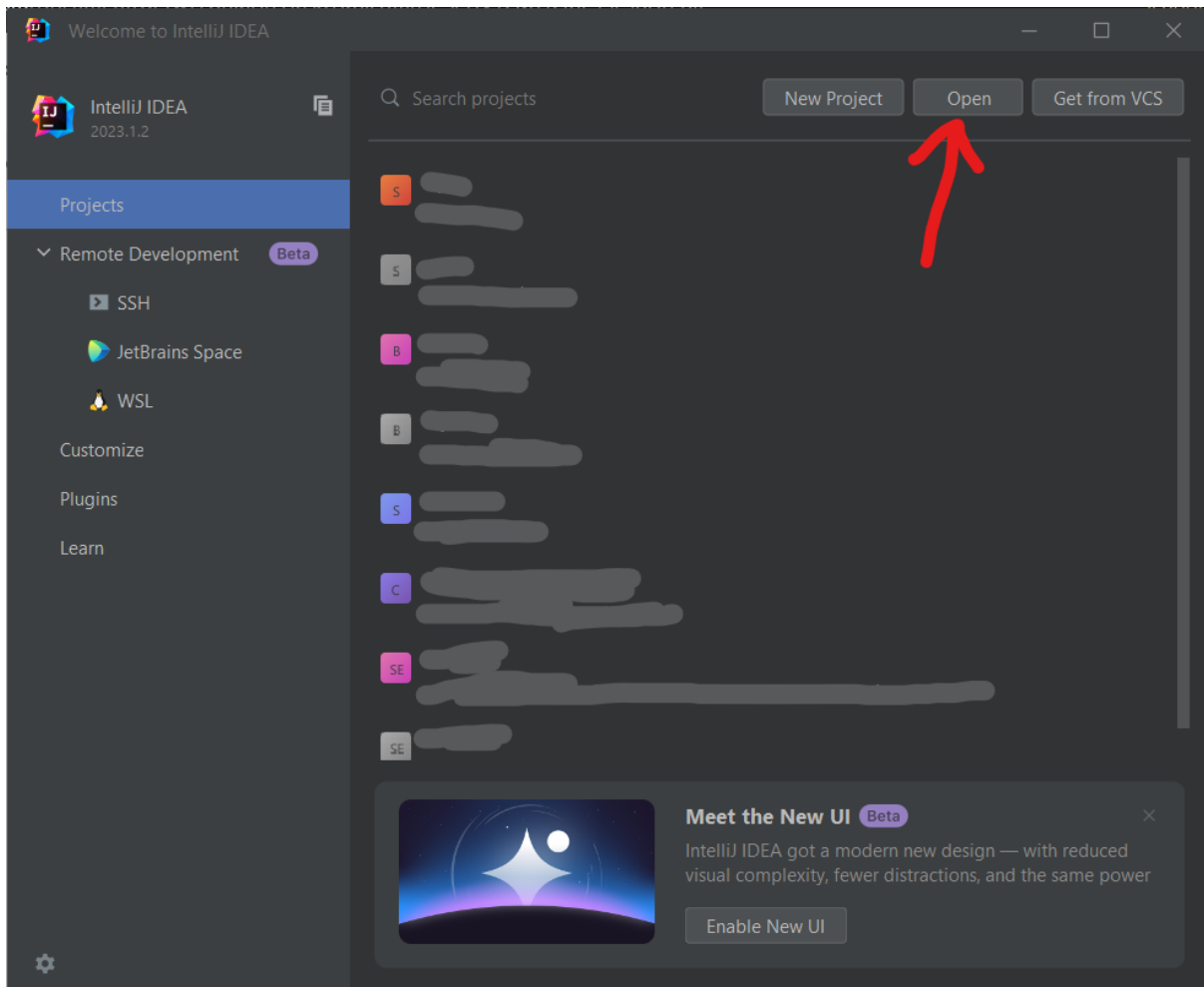
Username

Contraseña

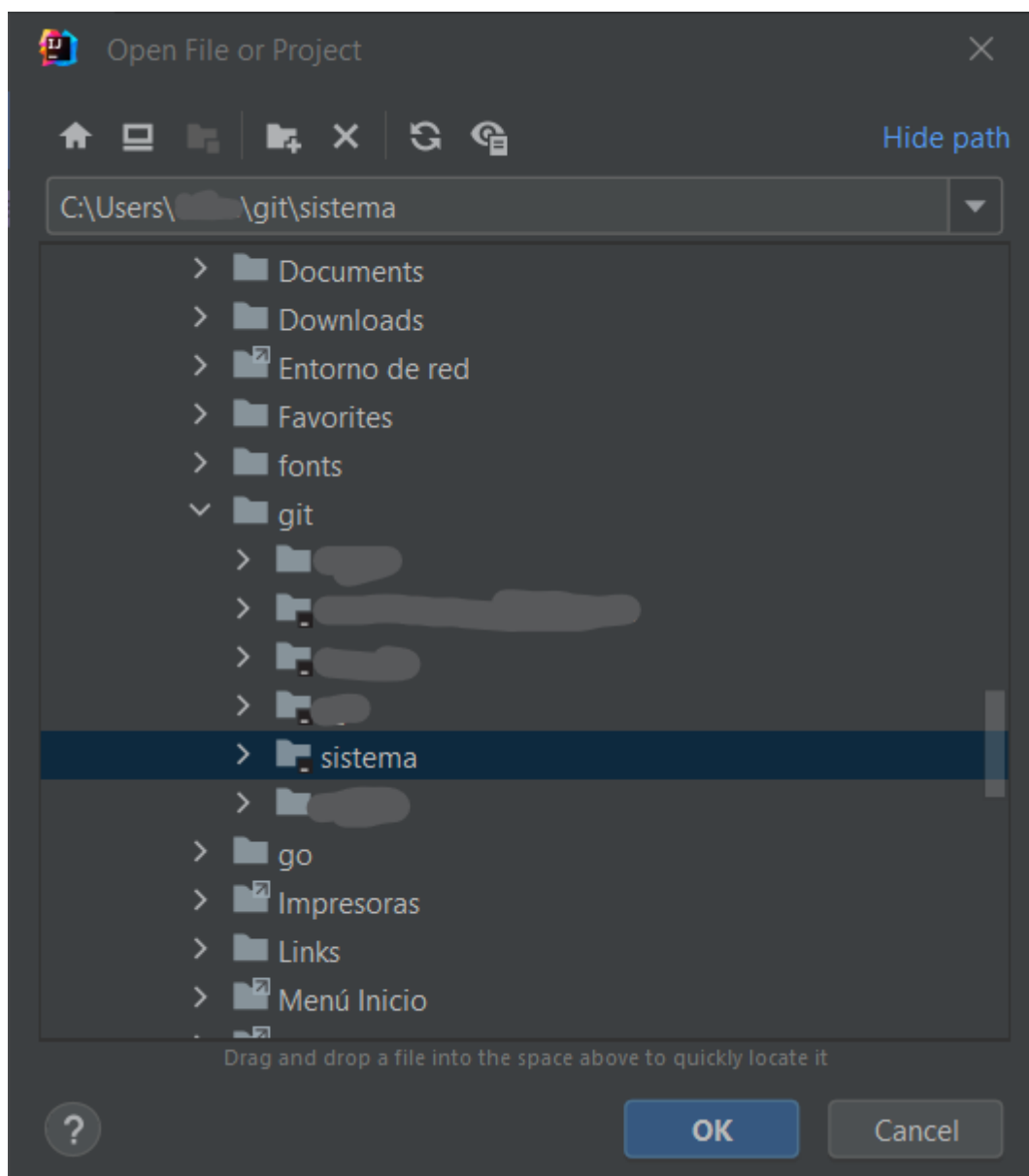
Password

Iniciar sesión

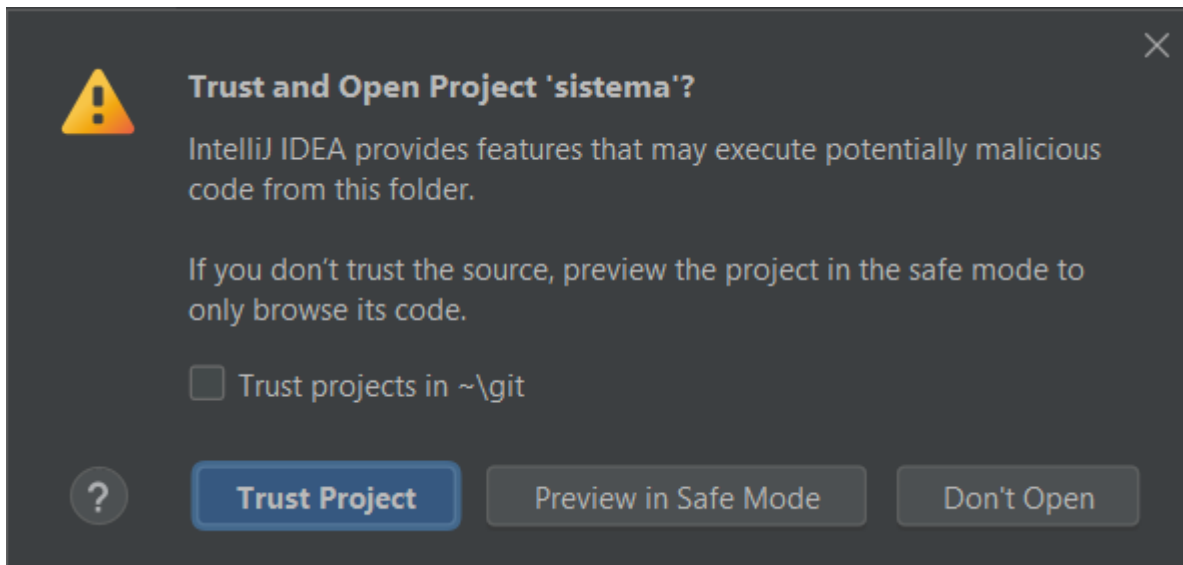
- 



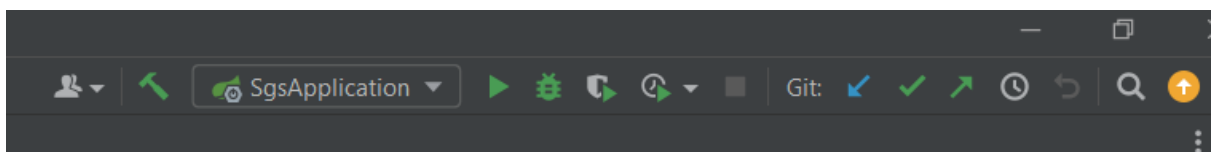
- 4.3. Navegar a la carpeta donde se clonó el repositorio y presionar “ok”.



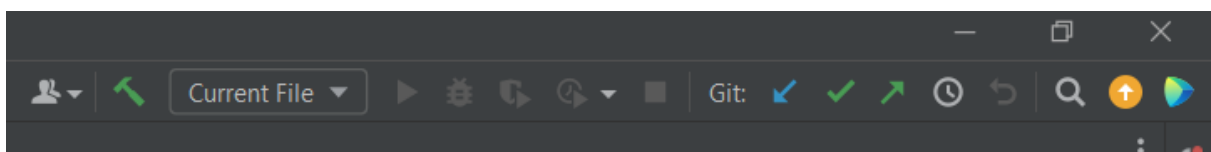
- 4.4. Si se presenta una pantalla como la siguiente, presionar “Trust Project”.



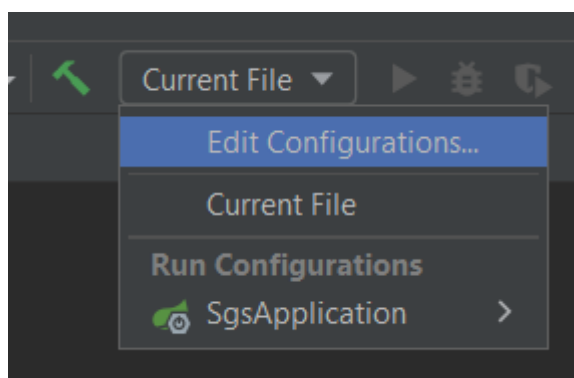
- 4.5. En la esquina superior derecha, debería aparecer un menú como el siguiente.



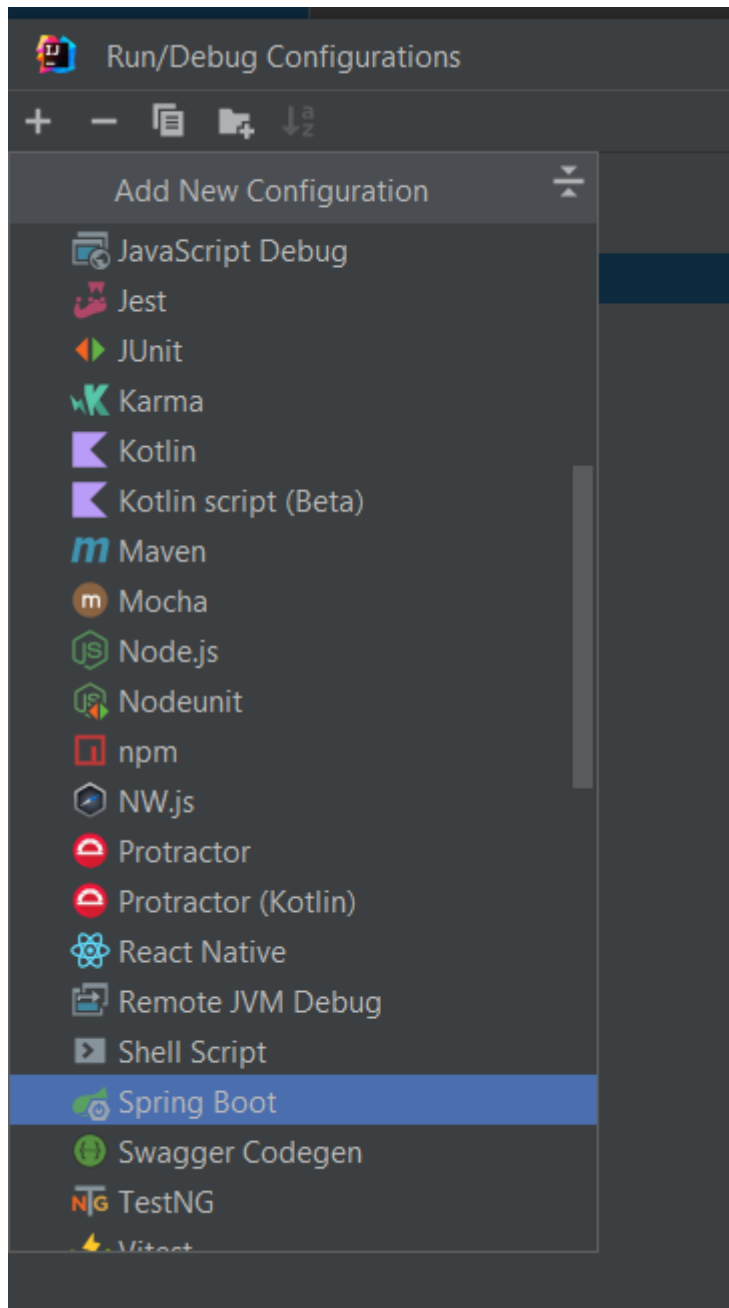
- 4.6. En concreto, debería mostrar SgsApplicaction, caso de que no se muestre así, y en su lugar aparece algo como lo siguiente, se deberá crear un perfil.



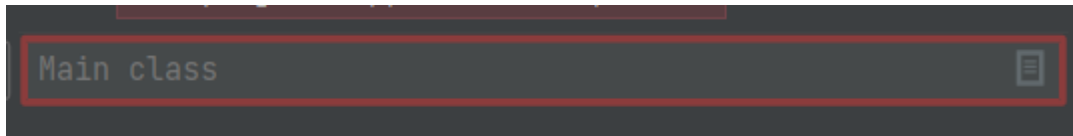
- 4.6.1. Se da clic en el menú mostrando "Current File". En el menú desplegado se da clic en "Edit Configurations".



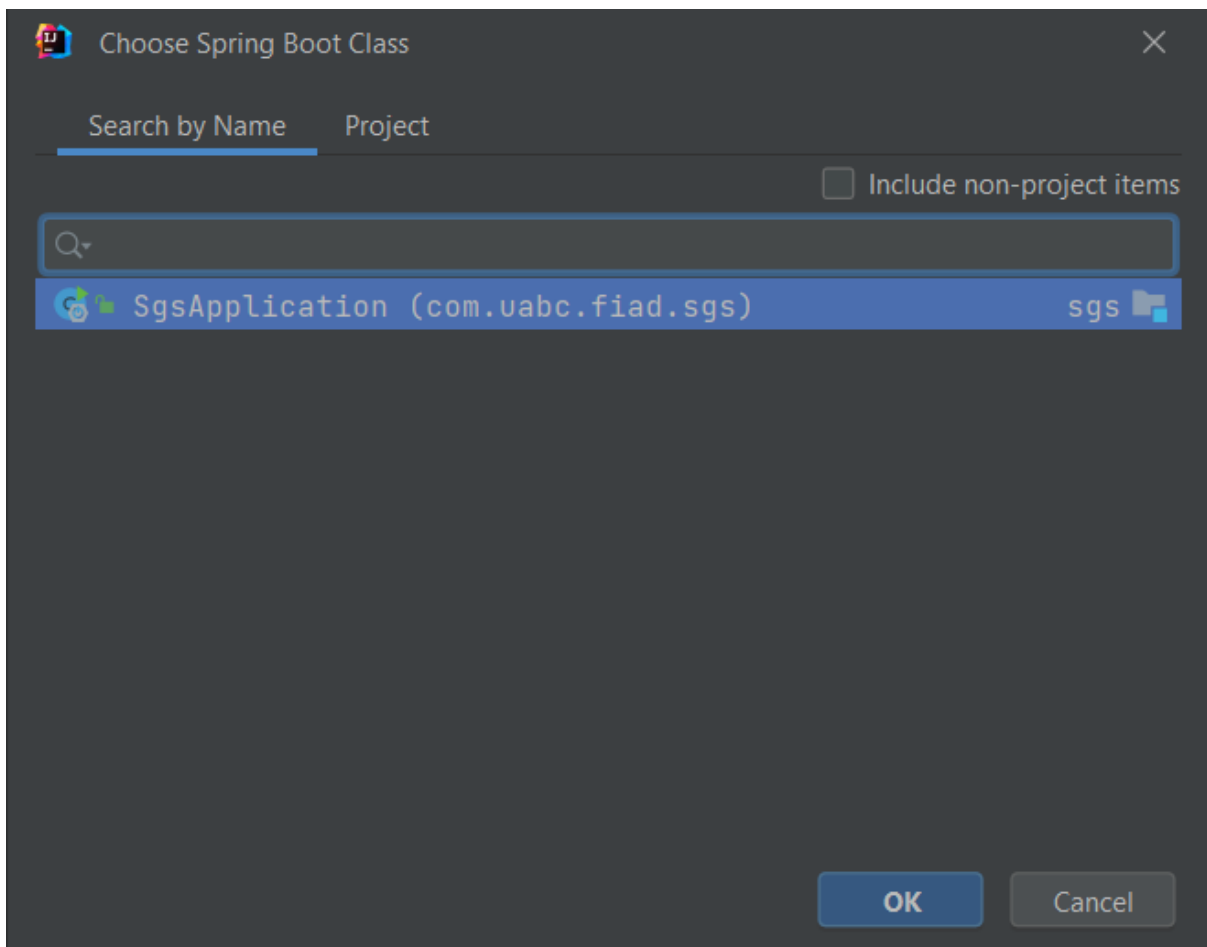
- 4.6.2. En la esquina superior derecha, presionar el símbolo de “+” y buscar la opción “Spring Boot”



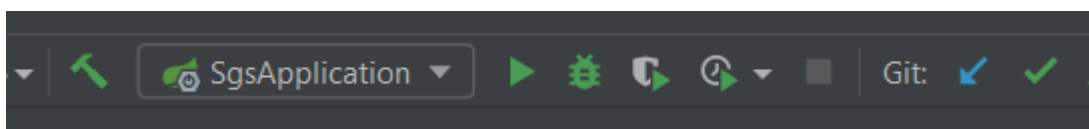
- 4.6.3. El nombre se puede usar el que sea, pero se recomienda "SgsApplication", y se usa la versión 17 de java.
- 4.6.4. Lo más importante es la clase, en el menú siguiente, se presiona el icono a la derecha del campo.



- 4.6.5. No debería haber más de una opción, pero en caso de que lo haya seleccionar la mostrada a continuación.



- 4.6.6. Presionar “ok” para seleccionar la clase, y otra vez “ok” para terminar de editar la configuración.
- 4.7. Finalmente, se presiona el botón de ejecutar al lado del menú de configuraciones.





4.8. Debería mostrar una salida similar a la siguiente.

```

  ____  _  __  ____ 
 / ___/  / /  /  _/  / ___/  ____  _  __  ____ 
/ /   /  / /  /  /  / /   /  ____  _  __  ____ 
/ /___/  / /  /  /  / /___/  ____  _  __  ____ 
\___/  / /  /  /  / \___/  ____  _  __  ____ 

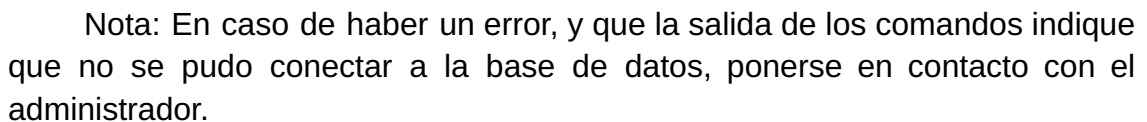
:: Spring Boot ::
          (v3.1.4)

2023-11-08T21:08:00.451-08:00 INFO 21116 --- [ restartedMain] com.uabc.fiad.sgs.SgsApplication : Starting SgsApplication using Java 17.0.7 with PID 21116 (C:\Users\...
2023-11-08T21:08:00.455-08:00 INFO 21116 --- [ restartedMain] com.uabc.fiad.sgs.SgsApplication : No active profile set, falling back to 1 default profile: "default"
2023-11-08T21:08:00.535-08:00 INFO 21116 --- [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : Devtools property defaults active! Set 'spring.devtools.add-properties'
2023-11-08T21:08:00.535-08:00 INFO 21116 --- [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : For additional web related logging consider setting the 'logging.level.
2023-11-08T21:08:01.621-08:00 INFO 21116 --- [ restartedMain] trationDelegate$BeanPostProcessorChecker : Bean 'org.springframework.ws.config.annotation.DelegatingWsConfiguratio
2023-11-08T21:08:01.659-08:00 INFO 21116 --- [ restartedMain] .w.s.a.s.AnnotationActionEndpointMapping : Supporting [WS-Addressing August 2004, WS-Addressing 1.0]
2023-11-08T21:08:02.150-08:00 INFO 21116 --- [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8090 (http)
2023-11-08T21:08:02.168-08:00 INFO 21116 --- [ restartedMain] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2023-11-08T21:08:02.252-08:00 INFO 21116 --- [ restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/] : Starting Servlet engine: [Apache Tomcat/10.1.13]
2023-11-08T21:08:02.253-08:00 INFO 21116 --- [ restartedMain] w.s.c.ServletWebServerApplicationContext : Initializing Spring embedded WebApplicationContext
2023-11-08T21:08:02.594-08:00 INFO 21116 --- [ restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1717 ms
2023-11-08T21:08:02.788-08:00 INFO 21116 --- [ restartedMain] o.s.b.a.w.s.WelcomePageHandlerMapping : Adding welcome page template: index
2023-11-08T21:08:03.168-08:00 INFO 21116 --- [ restartedMain] o.s.s.web.DefaultSecurityFilterChain : Will secure any request with [org.springframework.security.web.session.
2023-11-08T21:08:03.213-08:00 INFO 21116 --- [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : LiveReload server is running on port 35729
2023-11-08T21:08:03.225-08:00 INFO 21116 --- [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8090 (http) with context path ''
2023-11-08T21:08:03.225-08:00 INFO 21116 --- [ restartedMain] com.uabc.fiad.sgs.SgsApplication : Started SgsApplication in 3.232 seconds (process running for 4.681)
```

Nuevamente, la clave es la última línea donde se indica que se inició la aplicación.

Ver nota al final.

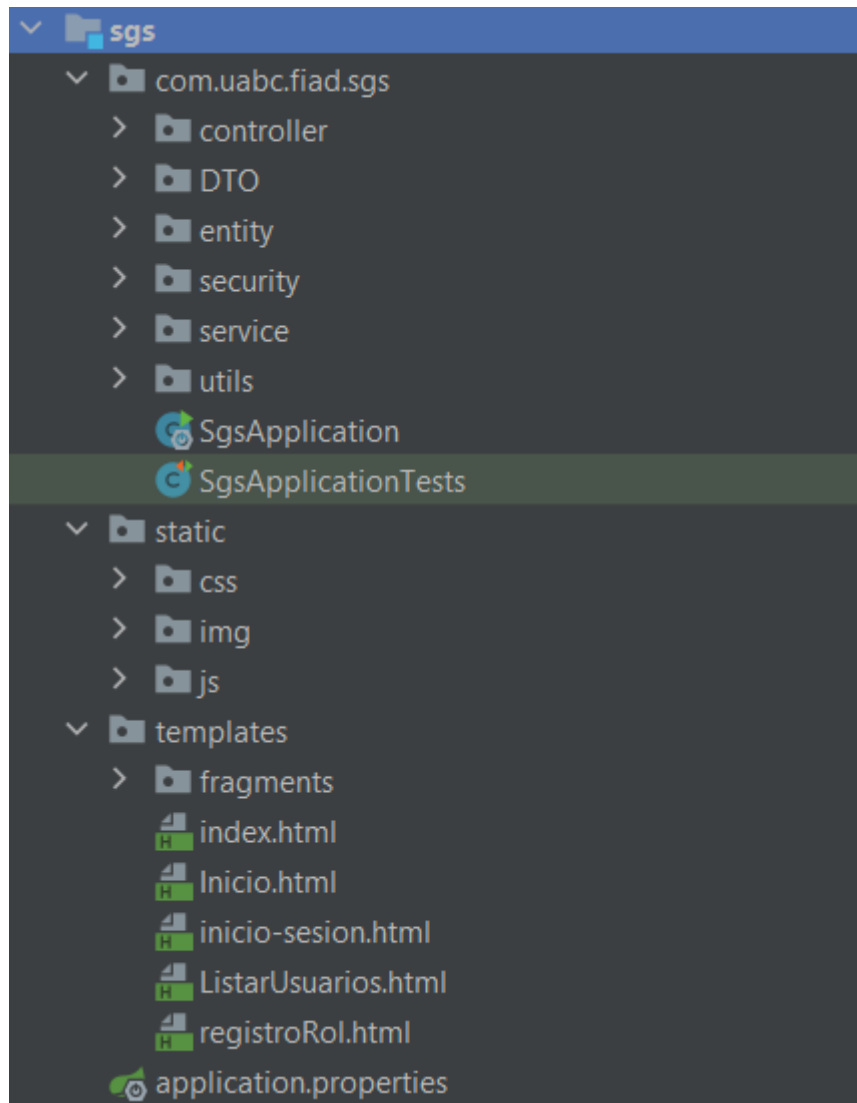
4.9. Para comprobar el funcionamiento, en un navegador entrar a la url <http://localhost:8090/>. Debería ver la página de inicio de sesión.



Lo que sigue son unos lineamientos para el desarrollo del proyecto, seguirlos es importante para asegurar una compatibilidad con el código existente y una mejor experiencia colaborativa.

Como podrá haber deducido, el sistema en cuestión usa Java y Spring Boot, en donde se decidió usar una arquitectura clásica de modelo, vista, controlador. En este manual no se va a entrar a detalle sobre la arquitectura, pero es importante tener esto en cuenta a la hora de programar.





Los paquetes tienen la siguiente función:

- controller
 - Todos los controladores.
- entity
 - Objetos representando tablas de la base de datos. Se prefieren usar objetos planos de Java (A veces conocidos como POJO), permitiendo usar anotaciones de lombok para facilitar su creación.
- security
 - Clases relativas a la seguridad, para la cual se usa la librería de Spring Security.
- service



- Para los servicios, usados para conectarse a la base de datos. Los servicios usan JDBCTemplate para conectarse a la base de datos y se comunican con el resto de la aplicación usando las entidades del paquete entity.
- utils
 - Utilidades generales compartidas en toda la aplicación.

Los recursos de javascript, como scripts o librerías, además de imágenes y archivos css, se almacenan en `src/main/resources/static`, y las plantillas html en `src/main/resources/templates`.

Librerías usadas

La librería más destacable es [htmx](#), la cual se usa en todo lugar donde se requiera actualizar información desde el servidor, ya sea en el perfil de usuario, listas de objetos de la base de datos, entre otros, y para subir formularios.

Más allá de eso, usamos bootstrap como librería de front-end y ninguna otra. Se tomó la decisión de no usar otras librerías de javascript, notablemente JQuery o React por dos principales motivos. El primero es mantener el código sencillo, agregar más librerías solamente complicaría el desarrollo, además de hacer la aplicación más pesada y por lo tanto lenta para los usuarios. La segunda razón lleva a una filosofía más general, manejar lo más posible por el servidor.

Si bien no se usa el servidor para hacer cosas triviales, los formularios funcionan bien para hacer que un campo sea obligatorio después de todo, se usa el servidor para todo lo demás. Un ejemplo perfecto es la validación del correo al momento de registrar un usuario. El formulario incluye, entre otras cosas, un input para el correo, el cual usa htmx para hacer una petición post, donde el servidor valida que sea válido de acuerdo a las reglas de negocio y regresa html plano (mediante un fragmento de thymeleaf pero también hay lugares donde regresamos html directamente) con el mismo input pero con formatos de error indicando si es inválido, o de éxito si es correcto.

Documentación en código y comentarios

Para aprovechar la potencia de las IDEs modernas, se optó por usar un estilo de documentación especial para una herramienta llamada [Javadoc](#). Se documentan todos los métodos de esta manera, permitiendo a la mayoría de



los IDEs modernos, incluyendo IntelliJ IDEA, mostrar la documentación de un método.

```
return getRegistrarUsuarioForm(u.getIdUsuario(), idRol: 0, perfil: true, model);
```

com.uabc.fiad.sgs.controller.UsuarioController

```
@GetMapping("/get-editar-form")
public String getRegistrarUsuarioForm(
    @RequestParam(value = "id") Integer id,
    @RequestParam(value = "idRol") Integer idRol,
    @RequestParam(value = "perfil", required = false) Boolean perfil,
    Model model
)
```

Params: id – el id del usuario a editar
perfil – bandera usada para entregar el formulario adecuado cuando se quiere editar el usuario en el modal de perfil
model – el modelo utilizado para pasar datos a la vista

Returns: fragmento para mostrar la información a editar

sgs

Esta documentación se usa para detallar el funcionamiento de los métodos en términos de alto nivel, centrándose en los datos que recibe, los efectos secundarios que provoca y los datos que entrega.

En cuanto a comentarios no hay un lineamiento específico, se deja a criterio del programador la manera en que prefiere documentar el código fuente, ya sea con muchos comentarios o código auto explicativo.