



Resultados del Proyecto

**Estandarización proceso
de desarrollo de software**

Gracias a la gestión de la investigadora principal, Mg. Ana Milena Rojas Calero, el proceso de desarrollo de software, explicado en detalle en el capítulo anterior, y todos sus artefactos resultantes de cada etapa fueron estandarizados en el sistema de gestión de calidad de la institución. En la siguiente tabla se relaciona cada artefacto, su descripción y código del documento por etapa

Etapa	Artefacto	Descripción	Código
Gestión de proyecto	Matriz de Riesgo de Proyectos	Documento donde se registra sprint a sprint los riesgos identificados y sus estrategias de control y mitigación.	GIF035 Matriz de Riesgo de Proyectos
	Matriz de Interesados	Documento donde se registran quienes son las personas involucradas en el proyecto, su influencia y sus datos de contacto.	GIF036 Matriz de Interesados
	Acta de Inicio de Proyecto	Documento donde se delimita el alcance del proyecto y se fijan los acuerdos que dan inicio a su ejecución.	GIF037 Acta delnicio de Proyecto SC
	Acuerdo de Confidencialidad	Documento legal que protege la información que la universidad expone a través del proyecto.	GIF040 Acuerdo de Confidencialidad DS
	Seguimiento Plan Informe Diario de Proyectos	Documento utilizado para calcular la velocidad del equipo y determinar así si se cumplirá con lo planeado o se requiere de algún ajuste.	GIF041 Segto Planife Inf. Diario de Proyectos
Análisis	Product Backlog	Documento utilizado para recopilar la información de tipo funcional asociada al proyecto a ejecutar.	GIF044 Historia de Usuarios y CriteriosDS
Diseño de software	Documento de Arquitectura de Software o DAS	Documento que contiene las decisiones arquitecturales que dan forma a la solución.	GIF038Dto.Arq.Software
	Manual de base de datos	Documento que describe las tablas construidas y sus relaciones.	ManualBDAUNIAJC_V1_1
Desarrollo	Código fuente y scripts BD	Son las unidades desplegables del producto.	Cómo tal, no tiene un código dentro del sistema, pero son el resultado final, que es puesto en funcionamiento dentro del servidor de Smart Campus una vez completado cada sprint. Es importante mencionarlo porque consolida el resultado de todo lo anterior.
Despliegue	Control de entregas de productos	Documento que contiene los resultados de la entrega del producto y sus recomendaciones.	GIF042ControldeEntregasde-ProductosDS

Configuración del ambiente de calidad preventiva

A continuación, utilizando el material elaborado en conjunto con los estudiantes William Andrés Leitón Muñoz y Brayan Alexis Moya Loaiza, para el trabajo de grado titulado: "Documentación e implementación de buenas prácticas DevOps con herramientas de automatización de pruebas y de despliegue a

través del refinamiento de software de Smart Campus”, dirigido por el profesor Manuel Alejandro Pastrana Pardo, se describe a continuación detalladamente bajo el enfoque del manual de instalación la configuración requerida para replicar el ambiente mencionado.

Instalación y configuración

Pasos previos

Previo a la configuración del ambiente de calidad preventiva, se requiere actualizar el sistema para evitar fallos de compatibilidad con las herramientas. Esto se realiza por medio de la instalación de paquetes complementarios que proporciona Enterprise Linux (EPEL) (EPEL, s.f.). El siguiente comando es utilizado para esta tarea:

- **sudo yum install epel-release**. Las figuras 39 y 40 reflejan los resultados de este comando.

```

developer@localhost ~$ sudo yum install epel-release
[sudo] password for developer:
Completamos cargados:fastestmirror
Determining fastest mirrors
 * base: mirror.ula.edu.ec
 * extras: centos.br.isanet.com.br
 * updates: centos.br.isanet.com.br
base                               | 3.6 kB | 00:00:00
extras                              | 2.9 kB | 00:00:00
updates                             | 2.9 kB | 00:00:00
1/21: centos/7/os_64primary_db      | 164 kB | 00:00:01
2/21: updates/7/os_64primary_db    | 6.7 kB | 00:00:00
Resolviendo dependencias
-> Ejecutando prueba de transacción
-> Paquete epel-release.noarch 8-7-11 debe ser instalado
-> Resolución de dependencias finalizada
Dependencias resueltas
=====
Package      Arquitectura  Versión      Repositorio  Tamaño
-----
Instalando:
epel-release
Sumen de la transacción
Instalar 1 Paquete
=====
Tamaño total de la descarga: 15 k
Tamaño instalado: 24 k
Is this ok [y/n]:

```

Figura 39. Actualización del sistema 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

```
delta RPM disabled because /usr/bin/applydelta not installed.
obsoletencia: /var/cache/yum/x86_64/epel/packages/epel-release-7-12.noarch.rpm: Encabezado00 RPM/00
56 Signature: ID de clave: 352c446c: NOKEY
No se ha instalado la llave pública de epel-release-7-12.noarch.rpm
1/23: epel-release-7-12.noarch.rpm                1 15 kB 00:00:00
1/23: sudo-1.8.23-4.el7_7.2.x86_64.rpm          1 842 kB 00:00:01
-----
Total: 429 kB/s | 656 kB 00:00:01
Resolviendo claves desde file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7
Importando llave GPG 8c35206415:
UsuarioID : "Fedora EPEL (7) (epel@fedoraproject.org)"
Huella    : 9167 7d7c 4d6e 98f1 713e 898f 682f a82c 352c 646f
Paquete   : epel-release-7-11.noarch (NoTras)
Reseña    : /etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7
Tipo de acuerdo: local
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Actualizando : epel-release-7-12.noarch                1/4
                sudo-1.8.23-4.el7_7.2.x86_64            2/4
Limpiendo    : epel-release-7-11.noarch                3/4
Limpiendo    : sudo-1.8.23-4.el7_7.1.x86_64            4/4
Comprobando  : sudo-1.8.23-4.el7_7.2.x86_64            1/4
Comprobando  : epel-release-7-12.noarch                2/4
Comprobando  : sudo-1.8.23-4.el7_7.1.x86_64            3/4
Comprobando  : epel-release-7-11.noarch                4/4
Actualizado:
  epel-release.noarch 0:7-12                sudo.x86_64 8:1.8.23-4.el7_7.2
!Listo!
desarrollero@localhost ~$
```

Figura 40. Actualización del sistema 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020) some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019)

Una vez completado el comando anterior, se procede a con la instrucción **sudo yum update** para actualizar todos los paquetes del sistema y una vez terminado, se reinicia el sistema con el comando **sudo reboot** para que se apliquen correctamente las modificaciones instaladas.

Herramientas necesarias para el proceso

UNZIP

En algunos casos, la información de los instaladores de las herramientas necesarias está en formato comprimido como el ZIP. Por lo anterior, es necesario instalar la herramienta UNZIP, que mediante un comando descomprime estos archivos. Así mismo, la herramienta nos permite también comprimir archivos si lo requerimos. Inicialmente se verifica si está instalado en el sistema con el siguiente comando: **unzip -v**. El resultado puede ser evidenciado en la figura 41.

```
UnZip 6.00 of 20 April 2009, by info-ZIP. Maintained by C. Spieler. Send
bug reports using http://www.info-zip.org/zip-bug.html; see README for details.

Latest sources and executables are at ftp://ftp.info-zip.org/pub/infozip/ ;
see ftp://ftp.info-zip.org/pub/infozip/UnZip.html for other sites.

Compiled with gcc 4.8.5 20150623 (Red Hat 4.8.5-39) for Unix (Linux ELF) on Aug 9 2019.

UnZip special compilation options:
  COPYRIGHT_CLEAN (PKZIP 0.9x unshrinking method not supported)
  SET_DIR_ATTRIB
  SYMBOLS (symbolic links supported, if RTL and file system permit)
  TIMESTAMPS
  UNISCRIP
  USE_EF_UT_TIME
  USE_UNSHRINK (PKZIP/Zip 1.x unshrinking method supported)
  USE_DEFLATED4 (PKZIP 4.x Deflated4(tm) supported)
  UNICODE_SUPPORT (wide-chars, char coding; UTF-8) (handle UTF-8 paths)
  MBCS_SUPPORT (multibyte character support, MB_CUR_MAX = 6)
  LARGE_FILE_SUPPORT (large files over 2 GiB supported)
  ZIP64_SUPPORT (archives using Zip64 for large files supported)
  USE_ZIP64 (PKZIP 4.6+, using help lib version 1.0.6, 6-Sept-2018)
  USE_TEXT_CONV
  (decompression, version 2.11 of 05 Jan 2007)

UnZip and ZipInfo environment options:
  UNZIP: (none)
  UNZIPOPT: (none)
  ZIPINFO: (none)
  ZIPINFOOPT: (none)
desarrollero@localhost ~$
```

Figura 41. Instalación de Unzip 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)

Si no se encuentra instalado, se procede a hacerlo con el comando: **sudo yum -y install unzip**. El resultado puede ser evidenciado en la figura 42.

```
Resolviendo dependencias
-> Ejecutando prueba de transacción
->> Paquete unzip.x86_64 0:6.0-29.el7 debe ser instalado
-> Resolución de dependencias finalizada
Dependencias resueltas
=====
Package      Arquitectura  Versión      Repositorio  Tamaño
Instalando:
unzip        x86_64        6.0-29.el7   base         170 k
=====
Resumen de la transacción
Instalar 1 Paquete
Instalar 1 Paquete
Tamaño total de la descarga: 170 k
Tamaño instalado: 362 k
Downloading packages:
unzip-6.0-29.el7.x86_64.rpm                1/170 KB 00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Instalando : unzip-6.0-29.el7.x86_64          1/1
Comprobando : unzip-6.0-29.el7.x86_64        1/1
Instalando:
unzip.x86_64 0:6.0-29.el7
¿listo?
[root@localhost data] #
```

Figura 42. Instalación de Unzip 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)

Finalizada la instalación de la herramienta, esta puede ser verificada mediante el siguiente comando: **which unzip**, como muestra la figura 43, que indicará la ruta donde quedó la herramienta configurada. Finalmente se verifica la versión para garantizar la instalación más actual con el comando: **unzip -v**.

```
[developers@localhost ~]# which unzip
/bin/unzip
```

Figura 43. Ruta de instalación de Unzip 3 en el sistema operativo CentOS 7.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020) some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019)

JDK (Java Development Kit)

El kit de desarrollo de java o JDK por sus siglas en inglés, permite la ejecución de algunas de las herramientas requeridas en la configuración indicada por Pastrana et al. (2019), por lo que su instalación en el sistema operativo resulta de gran importancia para el funcionamiento adecuado del ambiente de calidad. Por lo anterior, es necesario verificar si se encuentra o no instalado en el sistema operativo, utilizando el comando: **java -versión**. Si el JDK está instalado, el sistema operativo indicará la versión actual instalada como indica la primera parte de la figura 44. En caso contrario, dará una respuesta de que no reconoce el comando. Así mismo, en la segunda parte de la misma imagen, se evidencia que es posible que cohabiten varias versiones de java dentro del sistema operativo. Gracias al comando: **sudo /usr/sbin/alternatives -config java**, es posible seleccionar una en específico. A modo de ejemplo se selecciona la 1.8 (la única existente para el ejemplo de configuración).

```
developer@localhost ~$ java -version
java version "1.8.0_242"
OpenJDK Runtime Environment (build 1.8.0_242-808)
OpenJDK 64-Bit Server VM (build 25.242-b08, mixed mode)
developer@localhost ~$ sudo mvz-bin/alternatives --conf by java
(sudo) password for developers:
hay 1 programa que proporciona 'java'.
-----
Selección  Comando
-----
+ 1  java-1.8.0-openjdk.x86_64 /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.242-b08-0.e17_7.x86_64/jre/bin/java
+ /usr/bin/java
Presione Intro para mantener la selección actual(1), o escriba el número de la selección: 1
developer@localhost ~$
```

Figura 44. Versiones de Java disponibles.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

En caso de no estar instalado, es requerido utilizar el comando: **sudo yum install java-1.8.0-openjdk.x86_64** y, posterior a su ejecución, se verifica la ruta en la que queda instalado utilizando **which java** (recomendable copiar esta ruta, ya que es necesaria para otras configuraciones más adelante). El resultado de este comando se evidencia en la figura 45.

```
developer@localhost ~$ which java
/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.242-b08-0.e17_7.x86_64/jre/bin/java
```

Figura 45. Ruta del directorio Java.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

La mayoría de las herramientas que utilizan el JDK requieren de una configuración de una variable de entorno que permita referenciar donde se encuentra. Por lo anterior, se debe crear la variable de entorno **JAVA_HOME** en el **.bash_profile**, que es un archivo por lotes donde cada usuario tiene su configuración y es ejecutado cada vez que se inicia el sistema. Para acceder a este archivo, es necesario utilizar un editor de texto de línea de comandos como VI o NANO. En este caso se utiliza el comando: **sudo nano .bash_profile**. La figura 46 muestra el resultado de abrir el archivo y configurar la dirección donde está instalado el JDK, especificando la carpeta **/bin**, donde se encuentran los scripts ejecutables de instrucciones de esta herramienta. La instrucción configurada en el ejemplo es **JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.242.b08-0.e17_7.x86_64/**.

```
GNU nano 2.3.1 Fichero: .bash_profile
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs

JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.242.b08-0.e17_7.x86_64
PATH=$PATH:$HOME/bin:$HOME/.local/bin:$HOME/bin

export PATH
```

Figura 46. Ubicación de variable Java.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Así mismo, es necesario agregar la ruta absoluta de Java en el mismo archivo en la variable PATH (sin el /bin), para que el sistema detecte el Java y sea posible utilizarlo. La figura 47 muestra el resultado de esta configuración.

```
PATH=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.242.b01-8.el7.7.x86_64/jre/bin:$PATH
export PATH
```

Figura 47. Ubicación de variable Java.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Completado esto, se hace necesario recargar el **.bash profile** para que el sistema tome los cambios realizados, mediante el comando: **sudo source ~/.bash_profile**. Una forma rápida de validar si la variable de entorno quedó correctamente configurada es ejecutando: **echo \$JAVA_HOME**, que imprime la ruta contenida por la variable, como se ve en la figura 48.

```
echo $JAVA_HOME
/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.242.b01-8.el7.7.x86_64/jre
```

Figura 48. Variables de entorno 1.

Fuente: (Leiton Muñoz y Moya loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

PostgreSQL 9.2 o versiones más recientes

Para el ejemplo se tomó la versión 9.2, pero es recomendable utilizar versiones más actuales. Esta configuración es necesaria debido a que herramientas como sonarQube lo requieren para guardar el histórico de los análisis estáticos de código realizados. Esto se observará más adelante en las pruebas ejecutadas.

Debido a que por defecto los paquetes de instalación de esta base de datos no están en el sistema operativo, se deben adicionar utilizando el comando: **sudo yum install https://download.postgresql.org/pub/repos/yum/9.2/redhat/rhel-7x86_64/pgdg-redhat92-9.2-24.noarch.rpm -y**, para luego poder ser instalados como indica la figura 49.

```
~/pgdg-redhat96-9.6-3.noarch.rpm ~$
Completados cargados: 1 paquetes erróreo
pgdg-redhat96-9.6-3.noarch.rpm
Combinando /usr/lib/yum-root-ca/oidb/pgdg-redhat96-9.6-3.noarch.rpm: pgdg-redhat-repo-42.0-6.noarch
Rescando /usr/lib/yum-root-ca/oidb/pgdg-redhat96-9.6-3.noarch.rpm para ser instalado
Resolviendo dependencias
--> Ejecutando prueba de transacción
--> Paquete pgdg-redhat-repo.noarch.0:42.0-6 debe ser instalado
--> Resolución de dependencias finalizada

Dependencias resueltas
=====
Package      Arquitectura  Versión      Repositorio  Tamaño
-----
Instalando:  pgdg-redhat-repo      noarch      42.0-6      /pgdg-redhat96-9.6-3.noarch      7.5 k
Resumen de la transacción
-----
Instalar 1 Paquete
Tamaño total: 7.5 k
Tamaño instalado: 7.5 k
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction:
  Instalando : pgdg-redhat-repo-42.0-6.noarch      1/1
  Comprando  : pgdg-redhat-repo-42.0-6.noarch      1/1
Instalando:
pgdg-redhat-repo.noarch.0:42.0-6
El list:
root@localhost ~#
```

Figura 49. Añadiendo repositorio PostgreSQL.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Cuando el repositorio es reconocido por el sistema operativo, es posible instalar la base de datos mediante el comando: `sudo yum install postgresql92 postgresql92-server postgresql92-contrib postgresql92-libs -y`, como evidencia la figura 50.

```
Instalando : postgresql92-contrib-9.6.17-1PGDG.rhel7.x86_64      5/6
Instalando : postgresql-server-9.2.24-2.el7_7.x86_64            6/6
Comprando  : postgresql92-libs-9.6.17-1PGDG.rhel7.x86_64      1/6
Comprando  : postgresql-server-9.2.24-2.el7_7.x86_64            2/6
Comprando  : postgresql-libs-9.2.24-2.el7_7.x86_64             3/6
Comprando  : postgresql92-contrib-9.6.17-1PGDG.rhel7.x86_64   4/6
Comprando  : postgresql92-9.6.17-1PGDG.rhel7.x86_64           5/6
Comprando  : postgresql-9.2.24-2.el7_7.x86_64                  6/6
Instalando:
postgresql-server.x86_64.0:9.2.24-2.el7_7      postgresql92.x86_64.0:9.6.17-1PGDG.rhel7
postgresql92-contrib.x86_64.0:9.6.17-1PGDG.rhel7      postgresql92-libs.x86_64.0:9.6.17-1PGDG.rhel7
Dependencia(s) instalada(s):
postgresql.x86_64.0:9.2.24-2.el7_7      postgresql-libs.x86_64.0:9.2.24-2.el7_7
El list:
root@localhost ~#
```

Figura 50. Instalación PostgreSQL.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Para verificar que la instalación se realizó correctamente, se puede ejecutar el comando: `postgres --versión`, que indica la versión instalada de postgres como muestra la figura 51

```
(developers@localhost ~)$ postgres --version
postgres (PostgreSQL) 9.2.24
```

Figura 51. Verificación PostgreSQL.

Fuente. (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Para poder hacer uso de la base de datos instalada es necesario iniciar el servicio mediante el comando: `sudo postgresql-setup initdb`. La figura 52 muestra el resultado de ejecutar correctamente la instrucción.

```
root@localhost ~]# postgresql-setup initdb
Initializing database ... OK
```

Figura 52. Inicializando PostgreSQL.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Es recomendable habilitar PostgreSQL para que se inicie automáticamente cada vez que el sistema arranque, de lo contrario, el paso anterior habría que realizarlo cada vez que el sistema operativo se reinicie. Por tanto, las siguientes instrucciones pueden ser aplicadas:

- `systemctl enable postgresql`
- `systemctl start postgresql`

A modo de ayuda extra, se exponen una serie de comandos utilizados frecuentemente para manipular el servicio de PostgreSQL en caso de ser necesario:

- Verificar estado del servicio: `service postgresql status`
- Iniciar servicio: `service postgresql start`
- Detener servicio: `service postgresql stop`
- Reiniciar servicio: `service postgresql restart`

Cuando la base de datos ya se encuentra instalada y el servicio inicializado, se requiere configurar la contraseña del usuario postgres. El comando a utilizar es: `passwd postgres`. Los resultados de la ejecución de la instrucción pueden ser evidenciados en la figura 53.

```
postgres
root@localhost ~]# passwd postgres
Cambiando la contraseña del usuario postgres.
Nueva contraseña:
CONTRASEÑA INCORRECTA: La contraseña tiene menos de 8 caracteres
Vuelva a escribir la nueva contraseña:
passwd: todos los símbolos de autenticación se actualizaron con éxito.
```

Figura 53. Modificación de Password del usuario postgres

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Así mismo, es necesario realizar algunas configuraciones de seguridad para garantizar correcta conexión de las herramientas a la base de datos, de una manera segura. La primera configuración, consiste en habilitar la autenticación

md5, para que la base de datos solo pueda ser manipulada por usuarios con credenciales de acceso. Para esto se debe editar el archivo de configuración de permisos de la aplicación denominado `pg_hba.conf`. Usando nuevamente el editor NANO, se ejecuta el siguiente comando: `sudo nano /var/lib/pgsql/data/pg_hba.conf`. La figura 54 muestra cómo se debe ver el archivo al abrirlo por primera vez.

```
# TYPE DATABASE USER ADDRESS METHOD
# "local" is for Unix domain socket connections only
local all all peer
# IPv4 local connections:
host all all 127.0.0.1/32 ident
# IPv6 local connections:
host all all ::1/128 ident
# Allow replication connections from localhost, by a user with the
# replication privilege.
local replication postgres peer
host replication postgres 127.0.0.1/32 ident
host replication postgres ::1/128 ident
```

Figura 54. Configuración de autenticación antigua.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Dentro del archivo, en la columna **METHOD**, se debe cambiar los argumentos *peer* a *trust* y los *ident* a *md5*. como indica figura 55.

```
# TYPE DATABASE USER ADDRESS METHOD
# "local" is for Unix domain socket connections only
local all all trust
# IPv4 local connections:
host all all 127.0.0.1/32 md5
# IPv6 local connections:
host all all ::1/128 md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
local replication postgres peer
host replication postgres 127.0.0.1/32 ident
host replication postgres ::1/128 ident
```

Figura 55. Configuración de autenticación actual.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Realizada la edición del archivo y guardados los cambios, se debe reiniciar el servicio PostgreSQL para que aplique las configuraciones realizadas. Se recuerda hacer uso del comando: `sudo service postgresql restart`.

Finalmente, como se mencionó al inicio para la instalación de Sonar se requiere de una base de datos donde almacenar la información requerida por la aplicación. Por lo anterior, se recomienda crear la base de datos previo a la instalación. Esto se puede realizar mediante la instrucción SQL: `create database sonar owner postgres` como se ve la figura 56.

```
postgres=# create database sonar owner postgres;
CREATE DATABASE
postgres=# \l
          Listado de base de datos
+-----+-----+-----+-----+-----+-----+
Nombre | Dueño | Codificación | Collate | CType | Privilegio
+-----+-----+-----+-----+-----+-----+
postgres | postgres | UTF8         | es_CO.UTF-8 | es_CO.UTF-8 |
sonar    | postgres | UTF8         | es_CO.UTF-8 | es_CO.UTF-8 |
template0 | postgres | UTF8         | es_CO.UTF-8 | es_CO.UTF-8 | *c/postgres
template1 | postgres | UTF8         | es_CO.UTF-8 | es_CO.UTF-8 | postgres-C*/postgres
(4 filas)
```

Figura 56. Base de datos creada.

Fuente: (Leiton Muñoz y Moya Loaliza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Herramientas de calidad preventiva Smart campus, modelo actualmente implementado

El modelo expuesto por Pastrana et al. (2019) parte de la interacción que tienen los equipos de desarrollo con el versionador de código, garantizando la propiedad colectiva del código para garantizar un crecimiento evolutivo organizado del producto (Venters et al., 2018). A esta herramienta se suman dos, que son: el Jenkins que permite la integración continua y el analizador estático de código Sonar, que es disparado por la herramienta Jenkins cuando genera los empaquetados de la aplicación que está siendo versionada para verificar que los cambios realizados no dañan la unidad desplegable. Varios autores buscan esta sostenibilidad que garantice la calidad del incremento en todo momento, muy alineados de la mano de prácticas como las recomendadas por DevOps, como plantean Luz, Pinto, & Bonifácio (2019). A continuación, se detalla en orden las instalaciones y configuraciones de cada herramienta:

GIT

Por defecto, CentOS 7 ya tiene integrado los paquetes de instalación de la herramienta Git, por lo que sólo se requiere instalarlos usando el comando: `sudo yum install git`, como se evidencia en la figura 57.

```
perl-Fill-Temp          s390_64          0.23-0.1-3.e17         base          56 k
perl-Filter            s390_64          1.49-3.e17             base          79 k
perl-GitUtils          s390_64          2.48-3.e17             base          66 k
perl-Git               s390_64          1.0.3-1-21.e17_7      updates      525 k
perl-HTTP-Tiny         s390_64          0.033-3.e17            base          82 k
perl-PackTool          s390_64          3.48-5.e17             base          71 k
perl-Pod-Escapes       s390_64          1.1.0-2-294.e17_6     base          87 k
perl-Pod-Parser        s390_64          3.28-4.e17            base          216 k
perl-Pod-Simple        s390_64          1.63-3.e17            base          227 k
perl-Scalar-List-Utils s390_64          1.27-2-0.e17          base          236 k
perl-Socket            s390_64          2.010-4.e17           base          49 k
perl-Storable          s390_64          2.45-3.e17            base          77 k
perl-ThreadLocal       s390_64          2.38-28.e17           base          31 k
perl-ThreadLocalBkg    s390_64          1.03-0.e17            base          13 k
perl-Time-HiRes        s390_64          4.1.1-9920-3.e17      base          45 k
perl-Time-Local        s390_64          1.2380-2.e17          base          24 k
perl-ccanLint          s390_64          1.27-2.e17            base          13 k
perl-File              s390_64          4.5.10-3-294.e17_6   base          680 k
perl-macros            s390_64          4.5.10-3-294.e17_6   base          44 k
perl-powert            s390_64          1.0-200-0-0.e17       base          12 k
perl-podlators         s390_64          2.5.1-3.e17           base          112 k
perl-threads           s390_64          1.07-1.e17            base          33 k
perl-threads-shared    s390_64          1.43-6.e17            base          33 k
rpm-gpg                 s390_64          3.1.2-6.e17_6-1      base          494 k

Resumen de la transacción
-----
Instalar 2 Paquetes (+38 Paquetes dependientes)
Espacio total de la descarga: 16 M
Espacio instalado: 59 M
Is this ok [y/n]:
```

Figura 57. Instalación de Git.

Fuente: (Leiton Muñoz y Moya Loaliza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Para verificar la correcta instalación basta con aplicar el comando: **git--versión**, que debe mostrar un resultado similar al de la figura 58.

```
[root@localhost ~]# git --version
git version 1.8.3.1
[root@localhost ~]#
```

Figura 58. Versión actual de Git.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Una vez instalado el versionador, se requiere configurar el usuario y correo global, con el fin de poder autenticarse de manera remota. Para esto es necesario utilizar los comandos:

- `git config --global user.name "tu nombre"`
- `git config --global user.email "tucorreo@ejemplo.com"`

Para confirmar que las configuraciones han surtido efecto se debe ejecutar el comando: **sudo git config --list**, como muestra la figura 59.

```
[root@localhost ~]# git config --global user.name "baoya"
[root@localhost ~]# git config --global user.email "baoya17@hotmail.com"
[root@localhost ~]# git config --list
git: 'config' is not a git command. See 'git --help'.
Did you mean this?
config
[root@localhost ~]# git config --list
user.name=baoya
user.email=baoya17@hotmail.com
[root@localhost ~]#
```

Figura 59. Verificación de configuraciones.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Es recomendable instalar el paquete `libcurl-devel` mediante el comando: **sudo yum install curl-devel**. Esto se hace para evitar, como sucede en algunos casos, problemas con comandos como: `git clone`. Para que la biblioteca quede configurada es necesario ejecutar la sentencia `--with-curl`. El comando a ejecutar es siguiente:

- `sudo ./configure --prefix=/usr --with-curl`

SonarQube 7.1.

El analizador estático de código seleccionado, según el trabajo de Pastrana et al. (2019), para el Ecosistema Smart Campus es SonarQube. Lo primero que se debe realizar es obtener los paquetes necesarios para su instalación, con el siguiente comando:

- `sudo wget -O /etc/yum.repos.d/sonar.repo http://downloads.sourceforge.net/project/sonar-pkg/rpm/sonar.repo`

Descargados e incluidos los paquetes dentro del sistema operativo, se procede a hacer la instalación mediante el comando: `sudo yum -y install sonar`. Así mismo, una vez instalado, es necesario saber dónde queda configurada la herramienta para futuras configuraciones. Por lo tanto, para buscar la ruta se puede utilizar la siguiente instrucción: `sudo find / -name sonar`. El resultado de la ejecución del último comando puede ser evidenciado en la figura 60, que indica que el directorio de Sonar quedó ubicado en la ruta: `/opt/sonarqube-7.1` para este caso.

```

(developers@localhost ~)$ sudo find / -name sonar
/etc/rc.d/init.d/sonar
/var/lib/yum/repos/x86_64/7/sonar
/var/lib/jenkins/plugins/sonar
/var/lib/jenkins/plugins/sonar/META-INF/maven/org.jenkins-ci.plugins/sonar
/var/cache/yum/x86_64/7/sonar
/usr/bin/sonar
/opt/sonar
(developers@localhost ~)$
display all 148 possibilities? (y or n)
(developers@localhost ~)$
display all 148 possibilities? (y or n)
(developers@localhost ~)$

```

Figura 60. Búsqueda de directorio SonarQube.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Para facilitar su manipulación, es posible cambiar el nombre del directorio si así se desea. En este ejercicio se realiza el cambio del nombre del directorio `sonarqube-7.1` por `sonar` mediante el comando: `sudo mv /opt/sonarqube-7.1 /opt/sonar`. El resultado puede ser evidenciado en la figura 61.

```

[root@localhost data]# mv /opt/sonarqube-8.2.0.32929/ /opt/sonarqube
[root@localhost data]# cd /opt/sonarqube/
[root@localhost sonarqube]# ls
bin conf COPYING data elasticsearch extensions lib logs temp web
[root@localhost sonarqube]# _

```

Figura 61. Modificación del nombre de directorio SonarQube.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Una vez instalada la aplicación se debe proceder con las configuraciones respectivas de SonarQube para su adecuado funcionamiento. Para esto, es necesario manipular el archivo **sonar.properties**, con el editor de archivos nano usando el siguiente comando: **sudo nano /opt/sonar/conf/sonar.properties**. Dentro de este archivo, se deben realizar las siguientes modificaciones, en caso de que algunas de las instrucciones estén comentadas descomentarlas:

- Configurar el usuario y password de acceso a la base de datos. En este caso se debe utilizar el usuario y password de postgres configurado en los pasos previos. Para ello se deben modificar las líneas de la siguiente manera: **Sonar.jdbc.username=postgres** **Sonar.jdbc.password=admin**. La figura 62 muestra el archivo en el que se modifican las dos líneas anteriores.



Figura 62. Configuración de password y usuario de PostgreSQL en SonarQube.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Del mismo modo es necesario identificar y modificar el nombre de la base de datos por la que fue creada en los pasos previos con el nombre de sonar. La línea a modificada debe quedar así: **sonar.jdbc.url=jdbc:postgresql://localhost/sonar**, ver figura 63.



Figura 63. Modificación de nombre de base de datos PostgreSQL.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Sonar al igual que Jenkins requiere de una IP y un puerto específico para poder acceder a este servicio desde otras máquinas. Por defecto se ejecuta en el puerto 9000. Por tanto, es necesario adicionar este puerto o el que deseemos colocar disponible para este servicio, dentro de las configuraciones del firewall del servidor. Esto es posible mediante la siguiente instrucción: **sudo firewall-cmd --zone=public --permanent --add-port=9000/tcp**. Una vez realizado, es necesario reiniciar el firewall, con el comando: **sudo firewall-cmd --reload**, para que tome los cambios aplicados. La figura 64 refleja los cambios realizados.

```
firewall-cmd: error: argument --add-port: not allowed with argument --add-service
[root@localhost system]# firewall-cmd --zone=public --permanent --add-port=9000/tcp
success
[root@localhost system]# firewall-cmd --reload
success
[root@localhost system]#
```

Figura 64. Configuración de permisos de puerto 9000.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020) some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Adicionalmente, es necesario agregar las siguientes variables en el archivo de configuración **sonar.sh**:

- Ruta: **/opt/sonar/bin/Linux-x86-64/sonar.sh**
- Variable 1: **SONAR_HOME=/opt/sonar**
- Variable 2: **PLATFORM=Linux-x86-64**
- **WRAPPER_CMD="\${SONAR_HOME}/bin/\${PLATFORM}/wrapper"**
- **WRAPPER_CONF="\${SONAR_HOME}/conf/wrapper.conf"**

Es importante resaltar que es necesario configurar la versión de Java a utilizar dentro del Wrapper para que así la aplicación pueda ser ejecutada, ver figura 65. Esta configuración se realiza dentro del archivo **wrapper.conf**, que se encuentra en este caso en la ruta: **/opt/sonar/conf/wrapper.conf**. La línea a modificar bajo la configuración realizada en los pasos previos es:

- **wrapper.java.command=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.242.b08-0.el7_7.x86_64/jre/bin/java**

```
bin/wrapper.jar command-path bin/wrapper.jar  
bin/wrapper.jar command-path bin/wrapper.jar 1.0.0 openjdk 1.0.0.252 MSU 0.017 7.006 64-bit bin/wrapper.jar
```

Figura 65. Ruta del Java en la configuración del Wrapper.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Para que el sistema operativo tome todos los cambios realizados, Sonar debe ser reiniciado mediante el comando: **sudo service sonar restart**. El resto de la configuración de la aplicación Sonar se realiza directamente accediendo desde un navegador web. Esto abrirá un menú de instalación de las configuraciones básicas que finalmente lleva al ingreso de credenciales, las cuales por defecto son: admin para el usuario y la contraseña. Esto debe ser cambiado después de iniciar sesión la primera vez, por razones de seguridad.

Una vez ingresado en el sistema es posible crear proyectos para que sean analizados. Cuando se crea un proyecto la herramienta genera un token para identificar el proyecto a analizar, como indica la figura 66. Esta llave es fundamental para que Sonar pueda interactuar con otras aplicaciones como Jenkins de manera segura o con versionadores de código en la nube como bitbucket, github y gitlab.

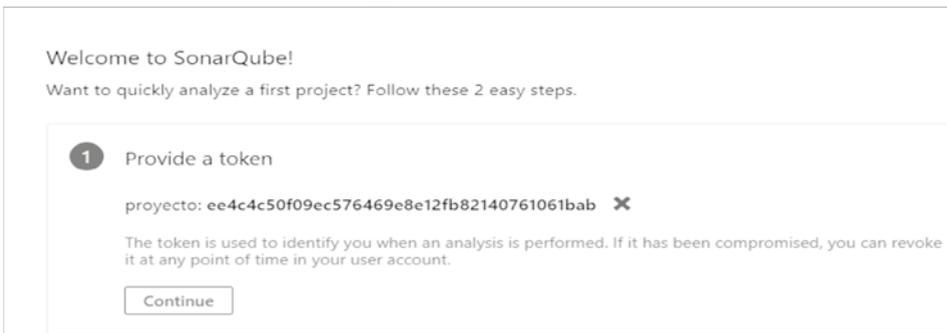


Figura 66. Inicio SonarQube.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Otro requisito importante cuando se crean proyectos, es que la herramienta solicita indicar el **lenguaje de programación del proyecto** que se analizará. Existen 3 opciones: Java, C# o VB.Net y otros. Así mismo solicitará la llave y el sistema operativo como indica la figura 67.

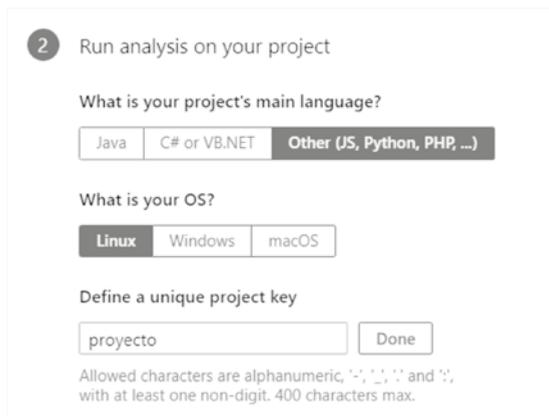


Figura 67. Configuración inicial de proyecto.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

La aplicación recomendará descargar sonar-scanner para el análisis de código, ver figura 68. Más adelante, se procederá con la instalación y configuración, por el momento en este punto se recomienda cerrar la ventana y continuar.



Figura 68. Requerimientos de SonarQube.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Adicionalmente, entre las configuraciones del proyecto también es importante **crear un usuario específico del proyecto** para, de este modo, no trabajar todo con el usuario administrador. Para realizar esto, se debe acceder a la *pestaña de Administración*, que se encuentra en el panel de navegación superior. Dentro de esta interfaz, seleccionar la opción *Seguridad e ingresar a Usuarios*. Ahí se debe presionar el botón azul de *Crear usuario* para realizar esta labor, como indica la figura 69.



Figura 69. Menú Usuarios en SonarQube.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020) some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

En esta pantalla se deben llenar los datos del usuario como muestra la figura 70.

The image shows the 'Crear usuario' form in SonarQube. It contains several input fields: 'Iniciar sesión *' with the value 'bmoya', 'Nombre *' with the value 'Brayan Moya', 'Correo electrónico' with the value 'bmoya17@hotmail.com', and 'Contraseña *' with masked characters. There is an 'Añadir' button for 'Cuentas SCM'. Below the form, a note states: 'El inicio de sesión y el correo electrónico se consideran automáticamente como cuentas SCM'. At the bottom right, there are 'Crear' and 'Cancelar' buttons.

Figura 70. Creación de Usuario en SonarQube.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020) some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Una vez realizado esto, es recomendable configurar los permisos globales de Sonar para entregar acceso granular a los roles, evitando que cualquier usuario pueda ejecutar ciertas tareas en los proyectos creados y solamente el administrador tenga permisos totales sobre la aplicación. Dentro de la misma interfaz de Administración, seleccionar la opción Seguridad y después *Permisos globales*. Ver figura 71. En el buscador se debe ingresar el usuario creado en el paso anterior y seleccionar los permisos que se le quieren dar.

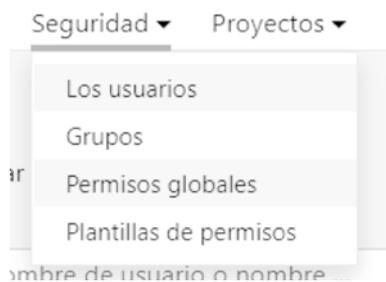


Figura 71. Configuración los permisos globales 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Con esto las configuraciones básicas de permisos quedan finalizadas y se puede proceder a revisar el proyecto. Para ello, es necesario dar clic en la opción *Proyectos* e ingresar a *Gestión de proyectos*. Dentro de esta interfaz, se observará el proyecto creado con su nombre, llave y último análisis (si tiene). Del lado derecho de la columna Último análisis hay un botón en forma de engranaje. Seleccionar y después ingresar en **Editar permisos**. Ver figura 72.



Figura 72. Configuración de permisos del proyecto 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Dentro de la interfaz de Permisos se debe colocar el proyecto como Privado, para que sólo usuarios con permisos de acceso a este proyecto puedan verlo. Luego en la opción Todos, dentro del buscador, se debe ingresar el usuario creado y darle los permisos correspondientes que tendrá en este en el proyecto, como indica la figura 73.

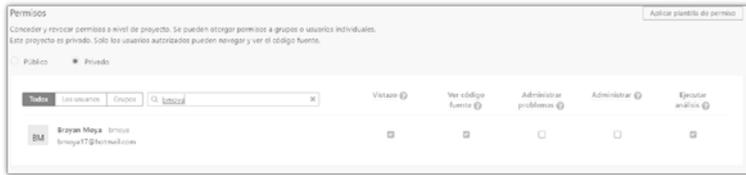


Figura 73. Configuración de permisos del proyecto 3.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020) some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

En caso de que Sonar no se inicie correctamente en el navegador, se puede intentar las siguientes posibles soluciones:

- Verificar que el estado del servicio con el comando: **sudo service sonar status**. Si no ha sido iniciado utilizar: **sudo service sonar start**
- Verificar que se haya configurado correctamente el archivo **properties**. Revisar las variables agregadas y la ruta bin de Java en el **wrapper.conf**. Modificarlas de ser necesario.
- Verificar que la IP sea la correcta al igual que el puerto en el que se ejecuta Sonar. Esto se puede revisar en el archivo properties de Sonar haciendo uso del editor de línea de comandos nano mediante la instrucción: **sudo nano /opt/sonar/conf/sonar.properties**, aquí revisar en la línea: **sonar.web.port=** y verificar que sea el mismo con el que se está intentando acceder. Si es una máquina verificar la IP que proporciona.
- Verificar si se encuentra abierto el puerto para acceso público. Si hay dudas con respecto a esto, ejecutar la siguiente línea: **sudo firewall-cmd --zone=public --query-port=9000/tcp**. Si el puerto está agregado, la línea de comandos indicará Yes. En caso contrario, se debe añadir el puerto al firewall de manera correcta, como se explicó al inicio de las configuraciones.
- Si el error persiste, verificar el archivo de log de errores de Sonar. Esto permite revisar otros errores no tan comunes al ejecutar el servicio. Cuando el servicio funciona correctamente, en el archivo se debe de ver una línea que indica SonarQube is up. El archivo puede ser editado con el comando: **sudo nano /opt/sonar/logs/sonar.log**

Sonar Scanner

Una vez configurado Sonar, es necesario instalar y configurar la herramienta Sonar Scanner para SonarQube que permite realizar el análisis estático

de código sobre los proyectos. Para instalar esta herramienta es necesario descargar mediante el siguiente comando el archivo empaquetado: **sudo wget https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-4.2.0.1873-linux.zip**. Para descomprimirlo usar el comando: **sudo unzip sonar-scanner-cli-4.2.0.1873-linux.zip**

Para que la herramienta funcione adecuadamente es necesario actualizar la configuración global para apuntar al servidor SonarQube configurado anteriormente. Esto se realiza editando el archivo de propiedades de Sonar Scanner con la instrucción: **sudo nano /opt/sonar-scanner-4.2.0.1873-linux/conf/sonar-scanner.properties**. Modificar como indica la figura 74.



Figura 74. Actualización de configuración global.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Adicionalmente, se deben agregar las variables de entorno de Sonar y Sonar Scanner en el PATH del sistema (figura 75), mediante los siguientes comandos:

- SONAR_HOME=/opt/sonar/bin/
- SONAR_SCANNER_HOME=/opt/sonar-scanner-4.2.0.1873-linux/bin/

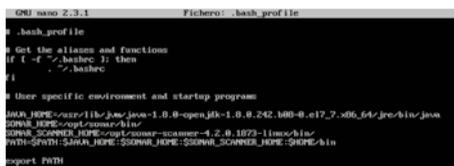


Figura 75. Configuración de las variables de entorno de Sonar y Sonar Scanner.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Al finalizar se deben guardar los cambios y luego recargar las configuraciones mediante el comando: **sudo source ~/.bash_profile**.

- Parar servicio: **sudo service jenkins stop**
- Reiniciar servicio: **sudo service jenkins restart**

```
(developers@localhost ~)$ sudo service jenkins start
Starting jenkins (via systemctl):
[ OK ]
(developers@localhost ~)$ sudo service jenkins status
jenkins.service - LSB: Jenkins Automation Server
Loaded: loaded (/etc/rc.d/init.d/jenkins; bad; vendor preset: disabled)
Active: active (running) since jue 2020-05-20 20:53:02 -05; 3min 25s ago
Docs: man:systemd-sysv-generator(8)
Process: 1127 ExecStart=/etc/rc.d/init.d/jenkins start (code=exited, status=0/SUCCESS)
Group: systemd.slice/jenkins.service
└─1586 /etc/alternatives/java -Dcom.sun.akuma.Daemon-daemonized -Djava.out.headless=tr...
may 20 20:52:55 localhost.localdomain systemd[1]: Starting LSB: Jenkins Automation Server...
may 20 20:52:55 localhost.localdomain rusermgr[1153]: pam_unix(rusermgr:session): session opened ...
may 20 20:53:02 localhost.localdomain jenkins[1127]: Starting Jenkins [ OK ]
may 20 20:53:02 localhost.localdomain systemd[1]: Started LSB: Jenkins Automation Server.
Hint: Some lines were ellipsized, use -l to show in full.
(developers@localhost ~)$ sudo service jenkins stop
Stopping jenkins (via systemctl):
[ OK ]
(developers@localhost ~)$
```

Figura 77. Estado del servicio Jenkins.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Jenkins utiliza por defecto el puerto 8080 al instalarlo por primera vez. Bajo el enfoque del top 10 de OWASP, trabajar con puertos por defecto no es recomendado, por tanto se cambia en esta configuración por el 9090. Para hacer el cambio basta con ingresar a editar el siguiente archivo de configuración **Jenkins** ubicado en la ruta: **/etc/sysconfig/Jenkins**. Su resultado puede ser evidenciado en la figura 78. El comando utilizado para editar el archivo es: **sudo nano /etc/sysconfig/Jenkins**.

```
jenkins@localhost:~$ nano /etc/sysconfig/Jenkins
# Whether to skip potentially long-running clean up at the
# ENDING type location. Do not enable this "true", unless
# you know what you're doing. See ENDING-25273.
#
ENDING_INITIAL_SKIP_CLEAN="false"
#
# Type: string
# Default: ""
# ServiceRestart: jenkins
#
# Options to pass to java when running Jenkins.
#
ENDING_JAVA_OPTIONS="-Djava.out.headless=true"
#
# Type: list
# Default: ["-Djava.net.preferIPv4Stack=true"]
# ServiceRestart: jenkins
#
# Don't Jenkins is listening on:
# Set to -1 to disable.
#
ENDING_PORT="8080"
#
# Type: string
# Default: ""
# ServiceRestart: jenkins
#
# IP address Jenkins listens on for HTTP requests.
# Default is all interfaces (0.0.0.0).
#
ENDING_LISTEN_ADDRESS=""
jenkins@localhost:~$
```

Figura 78. Cambio de puerto „8080“ por „9090“.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Para que los cambios surtan efecto, se requiere el reinicio del servicio Jenkins mediante el comando: **sudo service jenkins restart**, como muestra la figura 79.

```
root@localhost developers]# service jenkins restart
Restarting jenkins (via systemctl): [ OK ]
root@localhost developers]# _
```

Figura 79. Reinicio de Jenkins.

Fuente. (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Para hacer uso de Jenkins, es necesario invocarlo desde un navegador a través de la IP del servidor y el puerto disponible que permiten su acceso. Igual que en el caso de la configuración de SonarQube, se hace necesario permitir el tráfico entrante en el puerto que se configure autorizandolo en el firewall del servidor. En este caso, por cuestiones de seguridad, no se revelará el puerto del servicio real de Smart Campus y se trabajará con un puerto de ejemplo para la configuración. Los comandos utilizados y su resultado puede ser evidenciado en la figura 80:

- `sudo firewall-cmd --zone=public --permanent --add-port=8080/tcp`
- `sudo firewall-cmd --reload`

```
root@localhost ~]# firewall-cmd --zone=public --permanent --add-port=8080/tcp
Warning: ALREADY_ENABLED: 8080:tcp
success
root@localhost ~]# firewall-cmd --reload
success
root@localhost ~]# _
```

Figura 80. Permitir acceso de visitantes a Jenkins.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

La primera vez que se realiza el ingreso a la herramienta se pedirá la contraseña inicial de Jenkins, la cual se encuentra en la ruta: `/var/lib/jenkins/secrets/initialAdminPassword`. Nuevamente utilizando el editor de texto nano, se accede a la ruta con el siguiente comando:

- `sudo nano /var/lib/jenkins/secrets/initialAdminPassword`

Se ingresa al archivo, se copia la clave que aparece y se pega en el campo de Administrator password, como muestra la figura 81.



Figura 81. Prueba de Jenkins en navegador 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Luego de realizar el paso anterior, la aplicación redirige a un login donde el usuario y contraseña serán admin por defecto, ver figura 82.



Figura 82. Prueba de Jenkins en navegador 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

En caso de fallos al iniciar el servicio, se recomienda intentar algunas de las siguientes recomendaciones:

- Verificar el estado del servicio en el servidor con el comando: **sudo service jenkins status**
- En caso de no estar corriendo, iniciarlo mediante la instrucción: **sudo service jenkins start**
- Verificar si Java está instalado en el sistema operativo usando el comando: **java -version**
- Verificar que la IP pública proporcionada por el servidor sea la misma con la que se está ingresando en el navegador. Si es una máquina virtual, verificar la IP que proporciona.

- Verificar el puerto de acceso a Jenkins y que sea el mismo que se invoca desde el navegador. Recordar que esa configuración se encuentra en el archivo accesible con el comando: **sudo nano /etc/sysconfig/Jenkins** en la línea **JENKINS_PORT=**
- Verificar si el puerto que utiliza Jenkins está abierto. Ejecutar la siguiente línea: **sudo firewall-cmd --zone=public --query-port=9090/tcp**. Debe indicar como resultado yes, en caso contrario volver al paso en el que se añade el puerto a las reglas del firewall.
- Si el servicio no inicia ya habiendo ejecutado las anteriores posibles soluciones, se recomienda verificar el archivo log de Jenkins, y así controlar los errores que arroje. El archivo se encuentra accesible con el comando: **sudo nano /var/log/jenkins/jenkins.log**

Para realizar la configuración de plugins necesarios para integrar las herramientas previamente instaladas, se requiere primero ingresar al servicio de Jenkins desde el navegador y en la interfaz principal seleccionar *Administrar Jenkins*, que se encuentra en el menú al lado izquierdo como se muestra en la figura 83.

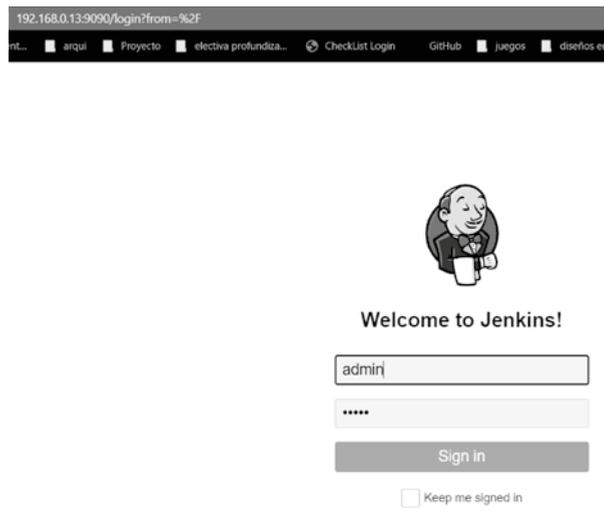


Figura 83. Instalación de plugins en Jenkins 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Dentro de la interfaz de *Administrar Jenkins* entrar a *Administrar plugins* como se ilustra en la figura 84:



Figura 84. Instalación de plugins en Jenkins 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Después ingresar a la opción, en lado derecho, parte superior, se encuentra un buscador. Introducir los nombres de los plugins que se desean descargar, en este caso GitHub y SonarQube Scanner para seleccionarlos, como se muestra en las figuras 85 y 86 respectivamente.

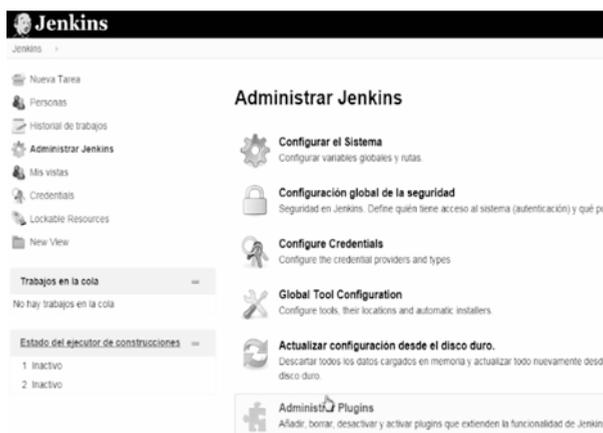


Figura 85. Instalación de plugin GitHub en Jenkins.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019



Figura 86. Instalación de plugin Sonar en Jenkins.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Adicionalmente, se instalarán tres plugin más. Uno es el plugin de **JUnit** para ver los reportes en formato XML, ver figura 87. Otro es el plugin de katalon, que se encargan de mostrar los reportes que proporciona **Katalon** (herramienta para automatización de pruebas) luego de su ejecución. Por último, el **Test Results Analyzer Plugin** sirve para generar reportes gráficos de la ejecución del proyecto de Jenkins.



Figura 87. Plugin JUnit.

Fuente: (Leiton Muñoz Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Teniendo los plugins seleccionados, presionar en *Descargar ahora e instalar después de reiniciar* para instalar los plugins, ubicado en la parte inferior como se muestra en la figura 88. Realizado esto se mostrará la interfaz de la figura 89 donde se visualizará el proceso de descarga e instalación de los plugins seleccionados. Seleccionar el campo *Reiniciar Jenkins cuando termine la instalación y no queden trabajos en ejecución*. Esto permite reiniciar automáticamente tanto el servicio de Jenkins como el navegador (ver figura 90).

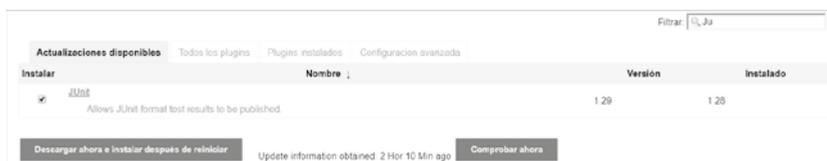


Figura 88. Instalación de plugins.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

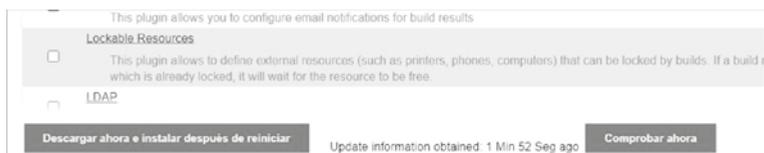


Figura 89. Instalación de plugins en Jenkins.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Instalando/Actualizando plugins

Preparación

- Probando conectividad con Internet
- Probando conectividad con jenkins-ci.org
- Correcto

Mailer

Descarga correcta. Se activará en el próximo arranque.

⇒ [Volver al inicio de la página](#)
(puedes empezar a usar los plugins instalados inmediatamente)

⇒ Reiniciar Jenkins cuando termine la instalación y no queden trabajos en ejecución

Figura 90. Reinicio de servicio y de navegador después de instalación de plugins.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Configuración de herramientas en Jenkins

Dentro de la interfaz de Administrar Jenkins, se selecciona la opción de *Configurar el sistema*. En la sección de *JDK*, se desactiva la casilla *Instalar automáticamente*, y se recomienda adicionar la que se configuró previamente. Para esto se debe dar un nombre (puede ser la versión que se esté usando) y se añade en el campo *JAVA_HOME* la misma ruta que se configuró anteriormente para el sistema como variable de entorno *JAVA_HOME*, ver figura 91. Además, se le adiciona el directorio *jre/bin/java* a la ruta indicada.



Figura 91. Configuración JDK.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

En la sección Git *desactivar la casilla Instalar automáticamente*. Se agrega la ruta del directorio bin de Git del sistema y se le da un nombre a esta, como se evidencia en la figura 92.



Figura 92. Configuración Git.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

En la sección de SonarQube Scanner, se selecciona *Añadir SonarQube Scanner* (ver figura 93), desactivando la casilla Instalar automáticamente y agregando la ruta donde quedó instalado Sonar (ver figura 94).



Figura 93. Configuración SonarQube Scanner.

Fuente: (Willian Andres Leiton Muñoz Brayan Alexis Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019



Figura 94. Configuración SonaQube Scanner 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

En la sección Instalaciones de SonarQube, se agrega la ruta donde esta instalado el servicio de Sonar, como indica la figura 95.

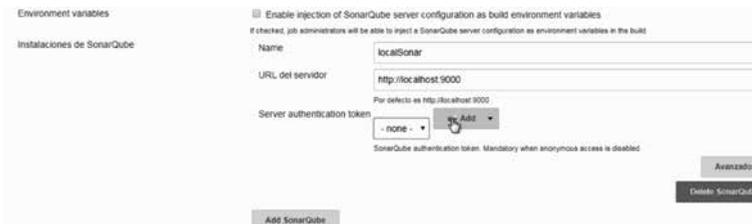


Figura 95. Configuración SonarQube Scanner 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Una vez dentro de esta configuración es necesario darle las credenciales con las que se inicia el servicio de sonar, ver figura 96.

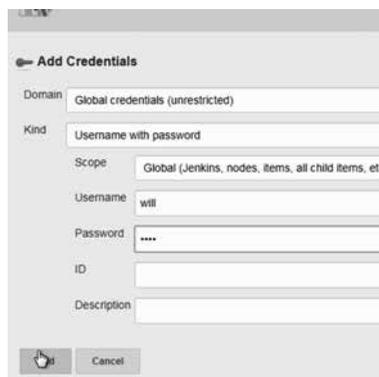


Figura 96. Configuración de SonarQube Scanner 3.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Terminado lo anterior, guardar los cambios realizados presionando el botón save, como muestra la figura 97.



Figura 97. Configuración de SonarQube Scanner 4.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020) some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Creando y configurando un proyecto

En este caso se realizará la configuración de ejemplo de un proyecto. Estos pasos servirán para cualquier tipo de proyecto, donde la única variación es el tipo de lenguaje utilizado de codificación. Para esto, en *la interfaz principal de Jenkins*, ingresar a la opción de *Nueva Tarea* ubicada en el menú, lado izquierdo, ver figura 98.



Figura 98. Configuración de Jenkins para proyecto de prueba en CentOS 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020) some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Dentro de esta opción se debe elegir el tipo de proyecto a crear, en este caso será un proyecto de estilo libre, debido a que se seleccionarán qué herramientas externas se integrarán y se tendrá un modo de ejecución personalizado. Adicionalmente, deber darle un nombre y presionar en el botón OK que se encuentra en la parte inferior izquierda, como se evidencia en la figura 99.



Figura 99. Configuración de Jenkins para proyecto de prueba en CentOS 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Una vez creado el proyecto, se habilitará un ítem con el mismo nombre que se ingresó en el paso anterior, dar clic en él como se muestra en la figura 100.



Figura 100. Configuración de Jenkins para proyecto de prueba en CentOS 3.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Esto abrirá una interface donde se debe ingresar a la pestaña *Configurar*, ubicada en el menú de lado izquierdo, ver figura 101. Esto permitirá la integración entre las herramientas previamente instaladas y su ejecución desde Jenkins.

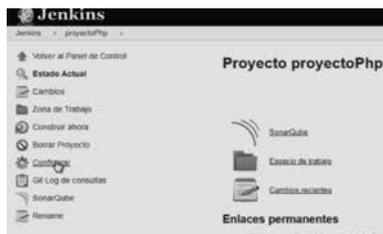


Figura 101. Configuración de Jenkins para proyecto de prueba en CentOS 4.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Dentro de estas configuraciones se debe dar clic en añadir (*Add*), y dentro de esta interfaz, como se puede ver en la figura 103. Aquí es necesario *configurar la dirección y las credenciales de acceso del repositorio Git*, donde está versionado el aplicativo, indicando la rama a la cual debe apuntar, en este caso master, como se evidencia en la figura 103.



Figura 102. Configuración de credenciales Git 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019



Figura 103. Configuración de credenciales Git 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Gracias a esto, el proyecto detectará cambios realizados en la rama del versionador y disparará automáticamente la ejecución de Jenkins para hacer las revisiones pertinentes de IC, y de las otras herramientas configuradas. Es importante resaltar que la herramienta permite realizar revisiones periódicas también. Para esto se debe configurar cada cuanto se realiza la revisión dentro de la sección *Disparadores de ejecuciones*. Es necesario seleccionar la casilla *Consultar repositorio (SCM)*, y aquí ingresar la expresión de tiempo para revisión, en este caso quedaría de la siguiente manera (H/5 * * * * *). Esto significa que cada 5 minutos el proyecto se ejecutará con un factor hash, evitando que todos los trabajos (Git, Sonar, Katalon y Reports) se ejecuten al mismo tiempo, ver evidencia en la figura 104.



Figura 104. Disparador de ejecuciones en Jenkins.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Así como fue configurado el repositorio, se hace la integración con el analizador estático de código SonarQube. Dentro de la sección *Ejecutar*, seleccionar *Añadir un nuevo paso* y seleccionar *Ejecutar SonarQube Scanner*, como se muestra en la figura 105.



Figura 105. Añadir un nuevo pasó en Jenkins.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Agregar el JDK previamente configurado y en la sección de *Analysis properties* insertar las siguientes líneas:

```
# Url del servidor donde se aloja sonar
sonar.host.url=http://192.168.0.22:9000

# Credenciales de acceso a la BD de SonarQube
sonar.jdbc.username=postgres
sonar.jdbc.password=admin

# Url de acceso a la BD
sonar.jdbc.url=jdbc:postgresql://localhost/sonar

# Credenciales de acceso al proyecto de SonarQube donde se muestran los resultados
```

```
del análisis
# ID único del proyecto
sonar.projectKey=proyecto
# Nombre del proyecto
sonar.projectName=proyecto

# Versión
sonar.projectVersion=1.0

# Dirección de la carpeta creada en Jenkins para el proyecto
sonar.projectBaseDir=/var/lib/Jenkins/workspace/prueba\ java

# Ubicación de la carpeta que contiene el código a analizar
sonar.sources=src

# Esta variable le indica a sonar que lenguaje será analizado, en caso de no especificar
alguno, se realizará el análisis con los lenguajes que tenga instalado
sonar.language=php

# Formato de codificación
sonar.sourceEncoding=UTF-8
```

La configuración deberá quedar como indica la figura 106:



Figura 106. Código de ejecución Sonar 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020) some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Se aplican y se guardan los cambios, para seguir con la instalación de Katalon y su configuración dentro de Jenkins.

Katalon

Antes de iniciar la instalación, es indispensable crear una cuenta en la página oficial (<https://www.katalon.com>) para poder utilizar la herramienta. Así mismo se recomienda revisar los requisitos del sistema y compatibilidad con los navegadores según (Katalon Studio: Entornos Soportados, s.f.), que se muestran en las Figuras 107 y 108, respectivamente.

Requisito	
SO	Windows 7, Windows 8, Windows 10, macOS 10.11+, Linux (basado en Ubuntu)
Procesador	Mínimo : 2 GHz o más rápido Procesador de 32 bits (x86) o 64 bits (x64)
Katalon Studio	Mínimo : 2 GB de RAM (32 bits) o 4 GB de RAM (64 bits) Recomendado : 4 GB de RAM (32 bits) u 8 GB de RAM (64 bits).
Katalon Runtime Engine	Mínimo : 2 GB de RAM (32 bits) o 4 GB de RAM (64 bits) Recomendación para ejecuciones concurrentes (y ejecución con Docker) : el número de sesiones concurrentes x 2 GB. Ejemplo : la RAM recomendada para 3 sesiones de ejecución simultánea es de 6 GB (3 x 2 GB).
Al menos 1 GB de espacio disponible en el disco duro. Se requiere espacio en disco adicional si	

Figura 107. Requisitos del sistema para Katalon.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Navegadores			
Navegadores de escritorio	Versión en Windows	Versión en macOS	Nota
explorador de Internet	9, 10, 11	N / A	Configuraciones de IE requeridas: Configuraciones de Internet Explorer.
Microsoft Edge	18 años	N / A	
Microsoft Edge (chrome)	80+	80+	Compatible con Katalon Studio versión 7.3+
Firefox	56+	56+	Para usar Firefox 57 con Katalon Studio, use Katalon Studio v5.1 +
Google Chrome	58+	58+	
Ópera	N / A	N / A	
Safari	N / A	12+	

Figura 108. Navegadores compatibles con Katalon.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Instalaciones adicionales para la correcta ejecución de las pruebas de Katalon

Como el servidor de Smart Campus utiliza el sistema operativo CentOS, que no posee interface gráfica y Katalon es una aplicación de entorno gráfico, es necesario instalar Chrome (o el navegador preferido permitido por Katalon, ver


```

Instalado:
  sory-x11-server-Xvfb.x86_64 0:1.20.4-10.e17

Dependencias instaladas:
  libXdmcp.x86_64 0:1.1.2-6.e17          libXfont2.x86_64 0:2.0.3-1.e17
  libXmu.x86_64 0:1.1.2-2.e17          libXt.x86_64 0:1.1.5-3.e17
  libXtst.i686.x86_64 0:1.0.9-3.e17    sory-x11-server-common.x86_64 0:1.20.4-10.e17
  sory-x11-south.x86_64 1:1.0.9-1.e17  sory-x11-xdb-qt11c.x86_64 0:7.7-14.e17

¡Listo!
    
```

Figura 111. Instalación de paquete xorg-x11-server-Xvfb.x86_64.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Instalación en Windows

Se usa la version de Katalon en Windows ya que, como se mencionó anteriormente, no se puede ejecutar la interfaz gráfica en el servidor. La versión en Windows se usa para crear las pruebas que posteriormente se configurarán dentro de Jenkins y finalmente son ejecutadas en el servidor CentOS 7. Además, según Katalon (s.f.), es importante tener en cuenta lo siguiente si se usa una versión inferior a la 7:

- No se podrán activar versiones inferiores a 7.
- No se actualizarán controladores.
- No habrá parches de seguridad y de rendimiento.
- No habrá mantenimiento futuro para la documentación, soporte o corrección de errores.

Por lo anterior, de la página oficial <https://www.katalon.com/download/> se realiza la descarga de una versión 7 o posterior (ver figura 112).



Figura 112. Instalación Katalon en Windows 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Esto baja un archivo comprimido, que debe ser extraído en una carpeta dentro del disco local C, donde se requiere ejecutar el archivo Katalon que se encuentra dentro de la carpeta. El paso a paso puede verse en las figuras 113 y 114, respectivamente.

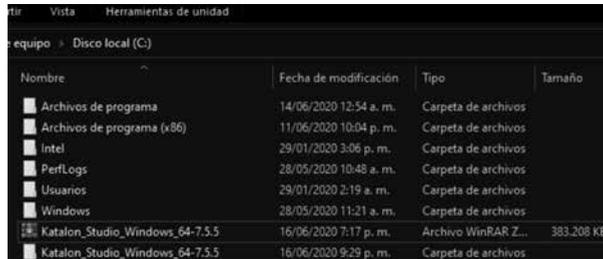


Figura 113. Instalación Katalon en Windows 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

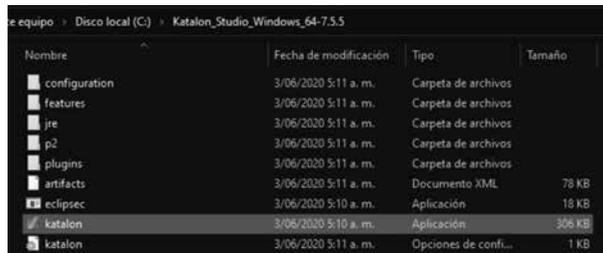


Figura 114. Instalación Katalon en Windows 3.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Creación de pruebas a modo de ejemplo

Esta sección busca explicar cómo unir Katalon y Git, ya que es una configuración general y puede variar muy poco, siendo útil para cualquier situación.

Al ejecutar por primera vez Katalon, saldrá una interfaz como la que se muestra en la figura 115, donde se debe ingresar el correo y contraseña que se registraron en la página oficial de Katalon (<https://www.katalon.com>), de esta forma se activará la aplicación para su uso.

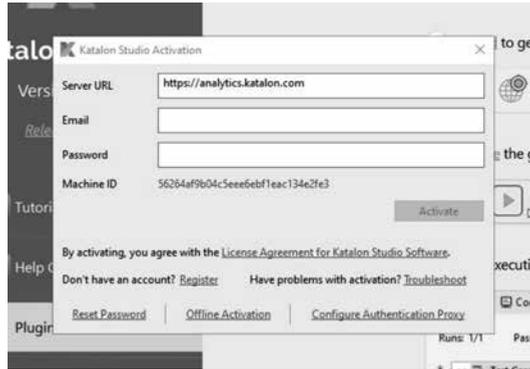


Figura 115. Login de Katalon.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Después de ejecutar Katalon, en el menú superior izquierdo se debe seleccionar la opción File y después, dentro del menú que despliega, dar clic en Project, como se muestra en la figura 116.

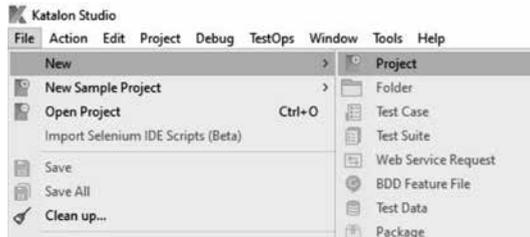


Figura 116. Nuevo proyecto en Katalon.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Aparecerá una ventana como la que se ilustra en la figura 117, donde se ingresa un nombre, tipo de proyecto, la plantilla que se usará, la URL del repositorio, la ubicación donde se guardará el proyecto y una descripción. El tipo de proyecto puede ser sólo Web o en combinación con una API, móvil o genérico. Para este ejemplo se selecciona el tipo Web y una plantilla en blanco (blank).

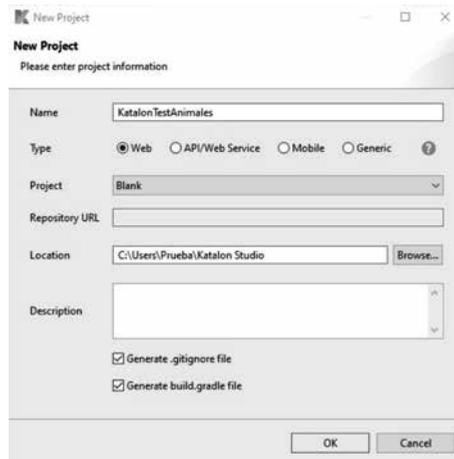


Figura 117. Configuración de nuevo proyecto en Katalon.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

En este caso se configura a Chrome como navegador por defecto para ejecutar las pruebas. Situarse en el menú superior izquierdo y desplegar las opciones de Project y dar clic en Settings; los pasos anteriores se muestran en la figura 118.

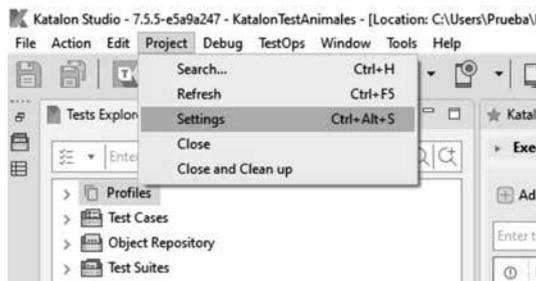


Figura 118. Navegador predeterminado Chrome en Katalon 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Para configurar el navegador es necesario en *Execution* del menú que se encuentra al lado izquierdo, cambiar las opciones dentro de *Default execution*, donde se muestran los navegadores. Ahí se puede seleccionar Chrome o el que se desee. Aplicar los cambios dando clic en el OK. Ver figura 119.

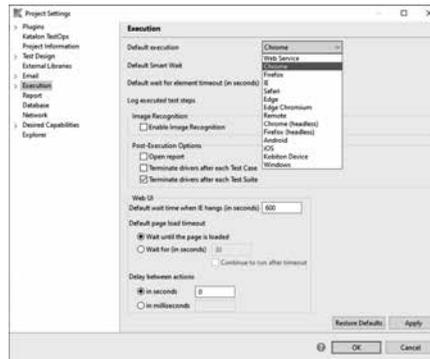


Figura 119. Navegador predeterminado Chrome en Katalon 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Para este ejemplo, se crearon tres Test Case y un Test Suite. Un *Test Case* o, por su traducción al español, *caso de prueba*, consiste en la grabación del paso a paso para cumplir una función específica, por ejemplo, el registro de un nuevo usuario. Esto creará automáticamente un script completo del proceso, donde se graba el paso a paso requerido para la transacción, diligenciando cada campo y presionando la opción de envío. En cambio, el *Test Suite* o, por su traducción al español, *conjunto de pruebas*, se basa en la ejecución de varios Test Case previamente configurados, que cumplen con varias funcionalidades asociadas a un proceso y que pueden tener o no dependencia entre sus resultados, por lo que se requiere que sean ejecutados en un orden.

Para comenzar con la creación del *Test Case*, dirigirse al menú izquierdo y desplegar las opciones de **Test Case**, después en **New** y seleccionar **Test Case**, como se indica en la figura 120.

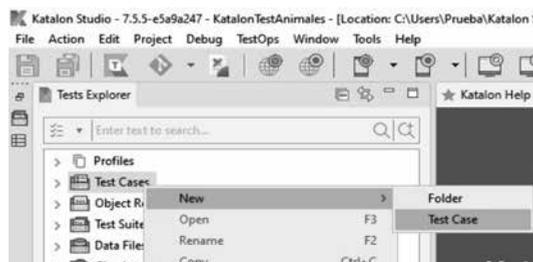


Figura 120. Configuración de Test Case en Katalon 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Esto desplegará una pantalla donde se ingresará el nombre y la descripción del caso de prueba, como muestra la figura 121. Se finaliza al dar clic en **OK**.

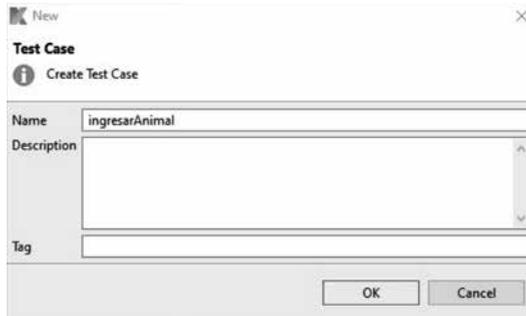


Figura 121. Configuración de Test Case en Katalon 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020) some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Hay tres formas de grabar los pasos. Forma manual (Manual Mode), Script (Script Mode) o por medio de la grabación y reproducción (Record & Play). El modo manual permite a los usuarios crear pruebas de automatización fácilmente con pocas habilidades de programación requeridas. El modo Script permite a los usuarios expertos escribir pruebas de automatización mediante programación en la vista Script de casos de prueba. El modo de grabación y reproducción consiste en grabar los pasos de una acción que se ejecuta en el navegador, después se creará un Script automáticamente para ser usado individualmente después o en un Test Case, haciéndolo más simple, para familiarizarse con la herramienta. En este caso se utiliza el modo de grabación y reproducción.

Para realizar una grabación, presionar el ícono en forma circular de color verde con un punto rojo (Record Web). Se recomienda ver la figura 122.

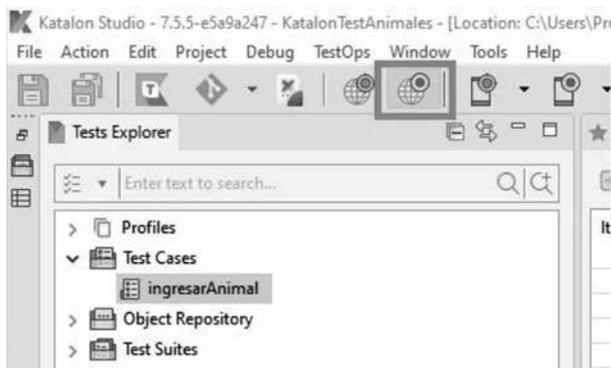


Figura 122. Configuración de Test Case en Katalon 3.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

A continuación, se desplegará una pantalla como en la figura 123, donde se ingresa la URL del aplicativo que se desea grabar y también se selecciona el navegador en el que se ejecutará el procedimiento.

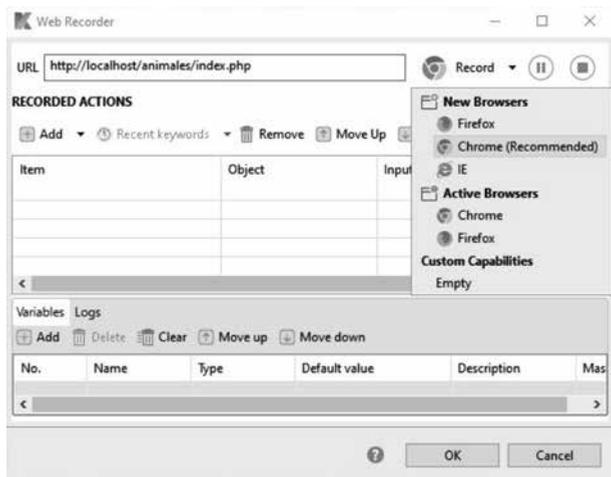


Figura 123. Configuración de Test Case en Katalon 4.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Para empezar la ejecución debe de dar clic en Record (figura 124), que ejecutará automáticamente el navegador, abriendo la aplicación a probar.

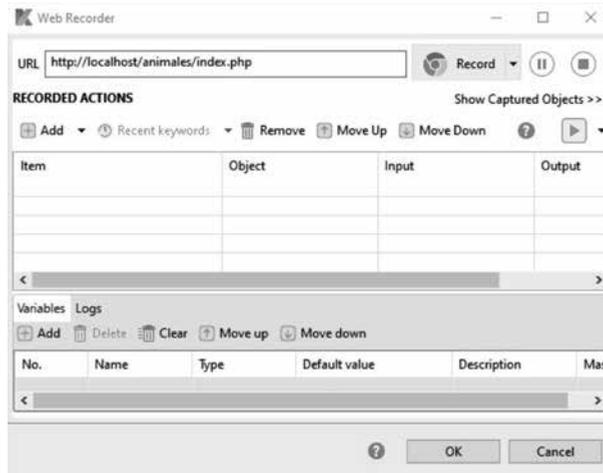


Figura 124. Configuración de Test Case en Katalon 5.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020) some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

En este caso se trata de un proyecto simple, que fue codificado con PHP puro y se grabará la acción de registrar un ítem (Nombre Animal), como se observa en la figura 125.

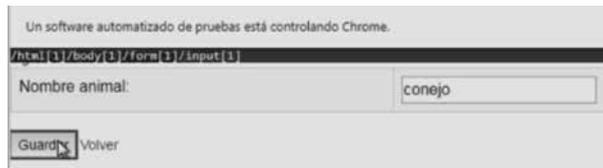


Figura 125. Configuración de Test Case en Katalon 7.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020) some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Para terminar la grabación de las acciones se cierra el navegador y automáticamente se abrirá una ventana. Ver figura 126 en la que se mostrará las acciones grabadas.



Figura 126. Acciones guardadas del Test Case en Katalon.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020) some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Una vez presionado **OK**, se desplegará la opción **Add Element to Object Repository** que se ilustra en la figura 127. Aquí se mostrarán todas las acciones grabadas y que quedarán almacenadas en una carpeta llamada **Object Repository**. Es posible crear una subcarpeta para especificar que las acciones realizadas son del módulo de ingresar un nuevo animal si uno lo requiere.

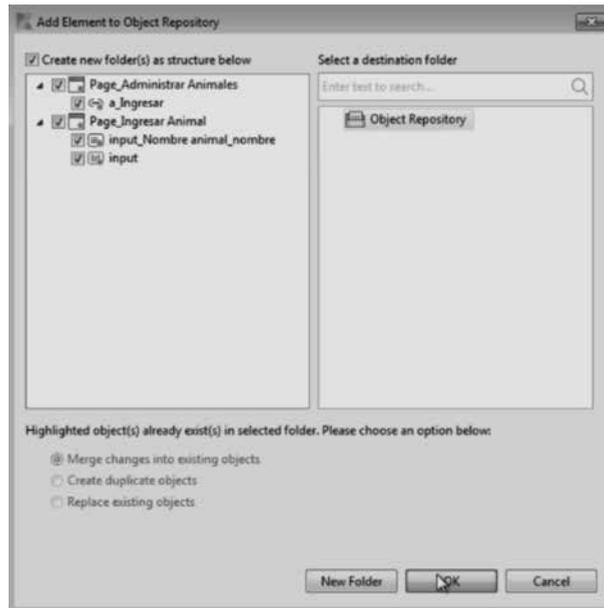


Figura 127. Repositorios de objetos en Katalon.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Al de dar clic en OK se mostrará el paso a paso grabado, como se ve en la figura 128.

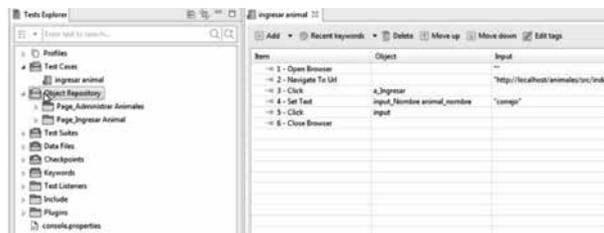


Figura 128. Interfaz principal de Katalon despues de añadir un Test Case.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Para continuar con el ejemplo, dentro del Test Suite, se adicionarán tres Test Case. En cada uno se grabará las acciones de Actualizar Animal y Eliminar Animal, como se hizo con Ingresar Animal. El resultado final del Test Suite organizado quedará como se muestra en la figura 129.

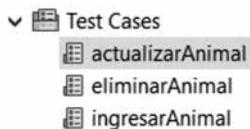


Figura 129. Creación de dos Test Case adicionales.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Para la configuración, en el menú izquierdo dar clic derecho sobre Test Suites para desplegar las opciones. Elegir **New** y, por último, **Test Suite**. Los pasos anteriores se ilustran en la figura 130

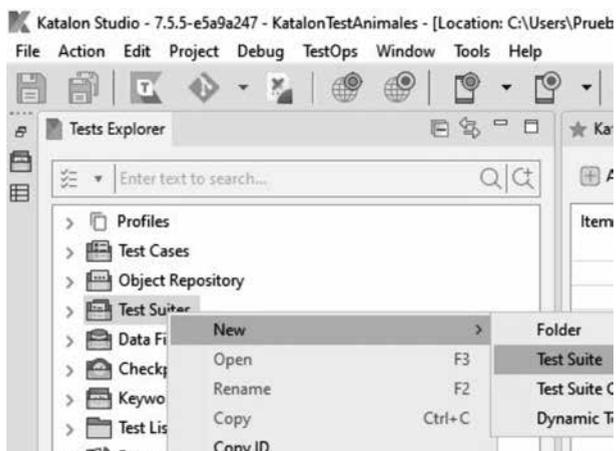


Figura 130. Creación de Test Suite en Katalon 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Después aparecerá una interfaz similar a la que se muestra en la figura 131, donde se debe ingresar un nombre y una descripción.

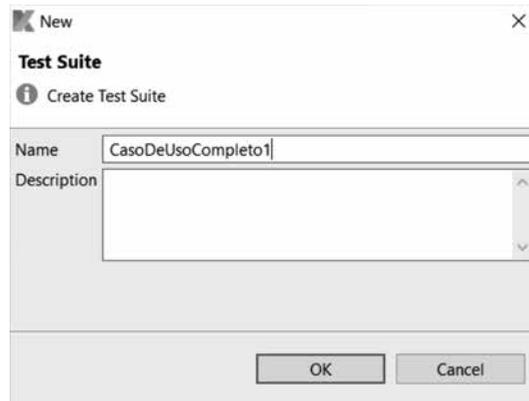


Figura 131. Creación de Test Suite en Katalon 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Lo anterior creará el Test Suite y se abrirá automáticamente una pestaña al lado derecho, en la cual se puede añadir los Test Cases creados anteriormente mediante la opción **Add**, ver figura 132.

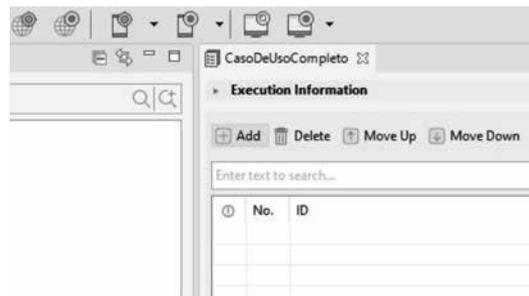


Figura 132. Creación de Test Suite en Katalon 3.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Es importante mencionar que al momento de añadir los Test Cases, se debe realizar en el orden en que se desea que se ejecuten. Para el ejemplo de guía, queda de la forma en que se muestra en la figura 133.

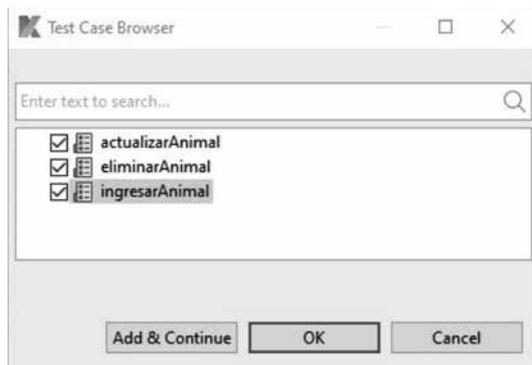


Figura 133. Creación de Test Suite en Katalon 4.

Fuente: (Leiton Muñoz Brayan Alexis Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Para ejecutar el Test Suite, se debe estar con el conjunto de pruebas abierto y seleccionado y dirigirse al menú de la parte superior central en la que se deben desplegar las opciones que se encuentran al lado derecho del logo de Play. Seleccionar el navegador en el que quiere ejecutar. Guiarse con la figura 134.



Figura 134. Ejecución de Test Suite en Katalon 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Automáticamente se abrirá una ventana del navegador seleccionado en donde se ejecutarán todas las acciones guardadas en el Test Suite, es decir, los Test Cases. El proceso de carga se muestra en la figura 135

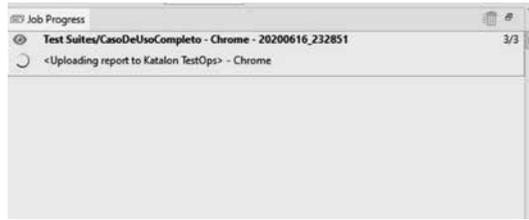


Figura 135. Ejecución de Test Suite en Katalon 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Los resultados se muestran mediante el log de ejecución, como indica la figura 136. Aquí se mostrarán cuáles Test Cases fueron ejecutados, el tiempo que tomó cada uno en ejecutarse y los resultados obtenidos.

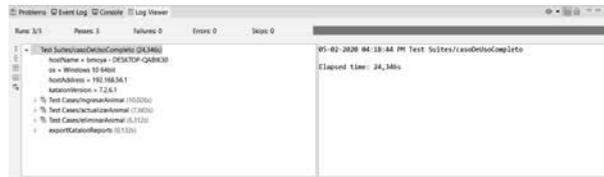


Figura 136. Log de ejecución de Test Suite en Katalon.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Luego comprobada su ejecución de manera correcta, se procede a *exportar como comando CMD* para poder implementarlo en Jenkins. Para ello, se debe ubicar el menú superior central y dar clic en el logo que contiene lo siguiente: los caracteres **>_ (Build CMD)**, resaltado en un recuadro rojo de la figura 137 para hacer más comprensible la explicación.

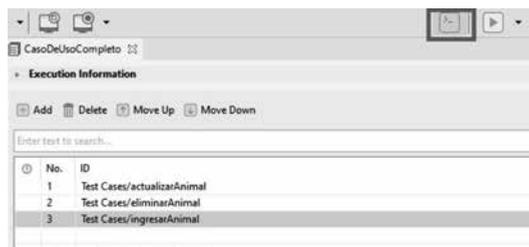


Figura 137. Comando CMD de Test Suite en Katalon 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Antes de continuar, para exportar el comando CMD es necesario generar una API Key de Katalon, que se podrá encontrar dentro del link <https://analytics.katalon.com/user/profile>, en donde se inicia sesión y después se ingresa a la pestaña de API Key para copiar el serial que se encuentra en el recuadro rojo, como se muestra en la figura 138.

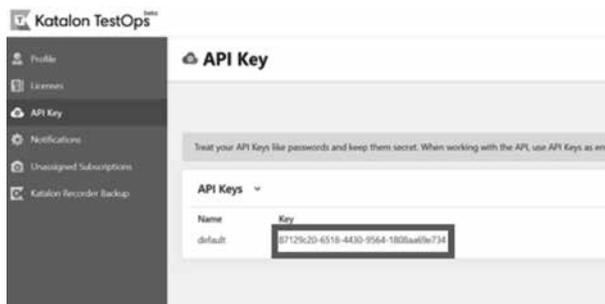


Figura 138. API Key de Katalon.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Dentro de la herramienta, se pega el **API Key** en el campo **Katalon API Key**, como se muestra en la figura 139.

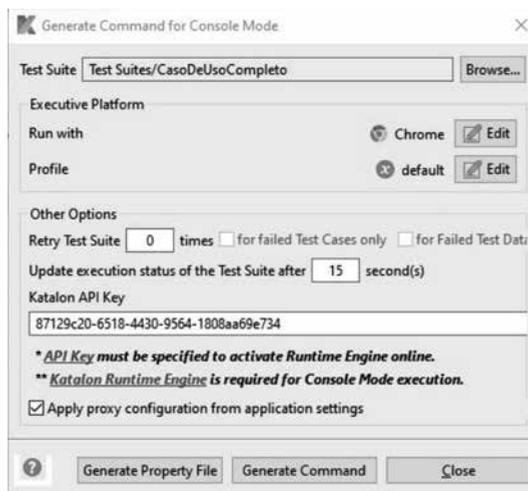


Figura 139. Comando CMD de Test Suite en Katalon 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Luego se debe seleccionar el Test Suite creado, por lo que dentro de la opción Browser se busca el que se ha creado anteriormente (ver figura 140).



Figura 140. Comando CMD de Test Suite en Katalon 3.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Por último, dar clic en **Generate Command**, como muestra en la figura 141. Este comando deberá copiarse para ser implementado dentro de la configuración del Proyecto en Jenkins.

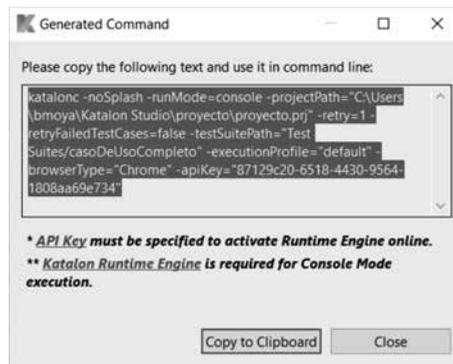


Figura 141. Comando CMD de Test Suite en Katalon 4.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Antes de comenzar con el proceso de las pruebas, es indispensable configurar Git para Katalon. Esto ayuda a versionar el conjunto de pruebas del proyecto, manteniendo el orden en todo momento. Por tanto, situarse en el menú superior izquierdo y dar clic en *Window* para desplegar las opciones (ver figura 142),

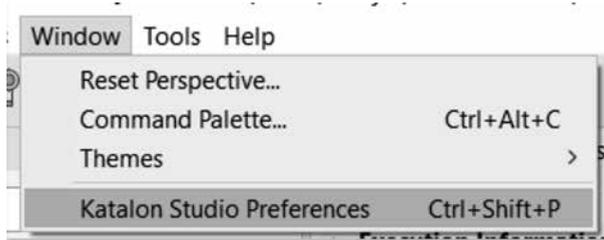


Figura 142. Configuración de Git en Katalon 1.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Después, se debe seleccionar *Katalon Studio Preferences* para abrir la interfaz que se muestra en la figura 143. Ahí seleccionar la opción *Git*, luego la opción *Enable Git Integration* y finalizar la configuración dando clic en *OK*.

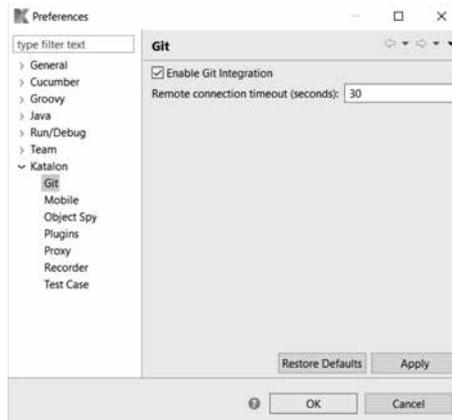


Figura 143. Configuración de Git en Katalon 2.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Al quedar configurado lo anterior, la herramienta habilitará un botón rojo con el logo de Git en la interfaz principal, ver figura 144. Este servirá para completar el resto de la configuración de Git.

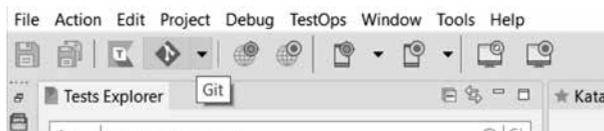


Figura 144. Configuración de Git en Katalon 3.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Desplegar las opciones con la fecha que se encuentra al lado derecho y dar clic en Share Project para iniciar Git, como se ilustra en la figura 145.

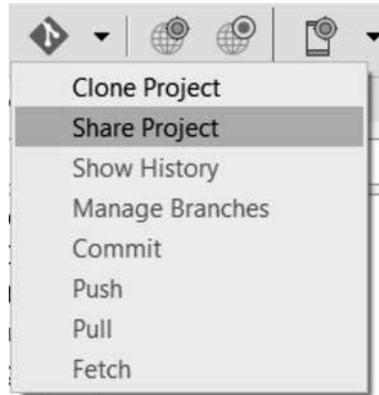


Figura 145. Configuración de Git en Katalon 4.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Si el resultado es exitoso saldrá un mensaje como el de la figura 146.

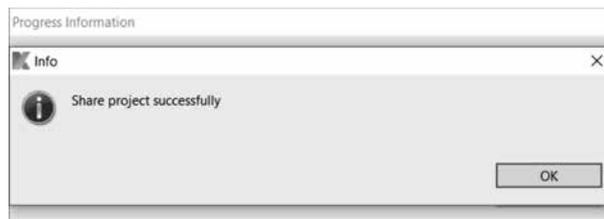


Figura 146. Configuración de Git en Katalon 5.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Se recomienda verificar que se haya creado la carpeta `.git` con el nombre del proyecto (animalesKatalon) en la carpeta Katalon Studio, que se encuentra en la ruta: `C:\Users\"usuario"\Katalon Studio`, como se ilustra en la figura 147 a modo de garantizar la correcta creación y vinculación del repositorio.

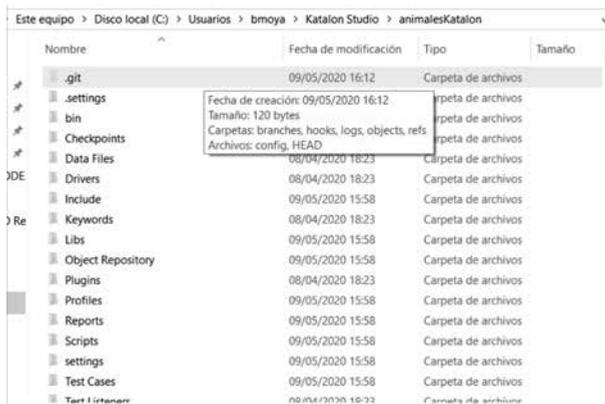


Figura 147. Configuración de Git en Katalon 6.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Una vez verificada la creación de la carpeta .git, se pasa a crear el repositorio directamente en Git (github.com), donde al final de la creación se proporciona una ruta de identificación de este repositorio creado, copiar ruta (ver figura 148) y tener en cuenta para el próximo paso.

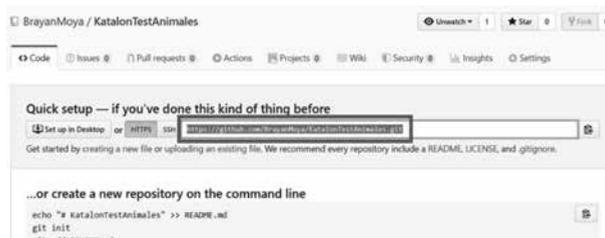


Figura 148. Configuración de Git en Katalon 7.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Luego, se es necesario hacer un Commit al Proyecto de Katalon para guardarlo en el repositorio creado en Git. Para tener más información acerca de las configuraciones en un proyecto, visitar este link oficial de Katalon: <https://docs.katalon.com/katalon-studio/docs/execution-settings.html#execution-settings>.

El proyecto creado anteriormente, como se encuentra en un repositorio de Git (ver figuras 149 y 150), se debe clonar dentro del servidor, para accederlo por parte de Jenkins. En este caso, se clonó dentro de la carpeta Workspace de Jenkins. Para hacerlo, se ingresa a la ruta donde se desea clonar y ejecutar

el comando *git clone* seguido de la ruta del repositorio, en este caso quedaría como se muestra en la figura 149:

```
leiton@leiton:~/Documents$ git clone https://github.com/leiton/katalon-test-framework.git
```

Figura 149. Ruta del repositorio.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Los archivos descargados se muestran en la figura 150:

```
leiton@leiton:~/Documents$ ls -la
total 12
drwxr-xr-x  7 leiton leiton  78 Jun 14 12:49 bin
-rw-r--r--  1 ??? root  426 Jun 7 16:58 build.gradle
drwxr-xr-x  2 leiton leiton   6 Jun 7 19:41 Checkstyle
-rw-r--r--  1 ??? root  134 Jun 7 16:58 compile.properties
drwxr-xr-x  2 leiton leiton   6 Jun 7 19:41 data files
drwxr-xr-x  2 leiton leiton   6 Jun 7 19:41 default
-rw-r--r--  1 ??? root   51 Jun 7 19:41 README
drwxr-xr-x  2 leiton leiton   6 Jun 7 19:41 resources
drwxr-xr-x  3 leiton leiton  51 Jun 14 12:54 lib
-rw-r--r--  1 ??? root  89 Jun 11 21:17 Object Repository
drwxr-xr-x  2 leiton leiton   6 Jun 7 19:41 TestRunner
-rw-r--r--  1 ??? root   26 Jun 7 16:58 testfiles
drwxr-xr-x  1 ??? root 1171 Jun 7 28:16 proyecto.prj
drwxr-xr-x  7 leiton leiton 167 Jun 14 12:49 src
-rw-r--r--  1 ??? root   69 Jun 11 21:17 testplan
drwxr-xr-x  4 ??? root   38 Jun 7 16:58 settings
-rw-r--r--  1 ??? root  181 Jun 11 21:17 test Cases
drwxr-xr-x  2 leiton leiton   6 Jun 7 19:41 TestRunner
-rw-r--r--  1 ??? root  186 Jun 11 21:17 Test Suite
```

Figura 150. Ruta y archivos de proyecto prueba.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Instalación y configuración en Jenkins

Katalon es la herramienta seleccionada para la automatización de pruebas del ambiente de calidad preventiva. Se podrá utilizar por medio de la inclusión de *Katalon Studio Plugin* dentro de *Jenkins*. Así, cuando un cambio en el código fuente es realizado a nivel del repositorio, la herramienta Jenkins detecta este cambio y dispara todo un ciclo de calidad a través de las herramientas que tiene integradas en el proyecto a revisar. En este caso, dispararía IC por medio de Jenkins; revisaría las pruebas unitarias del aplicativo, ejecutaría una revisión estática de código por medio de SonarQube y luego ejecutaría las pruebas funcionales automatizadas mediante Katalon. Cabe resaltar que esta forma de instalación se tiene que hacer sólo una vez y estará disponible de manera global para cualquier proyecto.

Para realizar la configuración de Katalon dentro de Jenkins, se debe ingresar a la interfaz de *Administrar Plugins* dentro de *Jenkins*. Buscar *Katalon Studio Plugin*, seleccionar para instalar (ver figura 151) y con esto quedará habilitado.



Figura 151. Instalación de plugin Katalon en Jenkins.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

En caso de no contar con el plugin, seleccionar la opción *Todos los plugins*, ingresar el nombre, seleccionar e instalar.

Una vez realizado esto, se vuelve a la opción de *Configuración* del proyecto y se debe ubicar la sección de *Ejecutar*. En esta sección dar clic en *Añadir un nuevo paso*, seleccionar *Execute Katalon Studio Test*, como se ve en la figura 152. Esto servirá para poder correr el script de pruebas Katalon generado anteriormente:

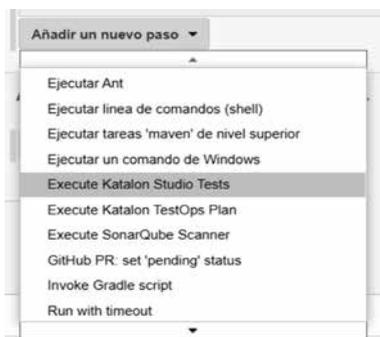


Figura 152. Paso nuevo para ejecutar pruebas de Katalon.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019

Posterior a esto, se debe seleccionar una versión de Katalon para instalar en el servidor CentOS en el campo *Download Katalon Studio Version* y, a continuación, pegar el comando generado de Katalon (CMD) (ver figura 163), pero ya configurado para CentOS. La variable *ProjectPath* que da el comando generado de Katalon es la que indica dónde está ubicado el archivo .prj a ejecutar, la ruta se cambiará y se colocará la que apunta a este mismo archivo del proyecto de Katalon, anteriormente clonado del repositorio. El código completo deberá quedar de la siguiente manera:

```
-noSplash --config -webui.autoUpdateDrivers=true  
-sendMail="bmoya17@hotmail.com" -projectPath="/var/lib/  
jenkins/workspace/prueba java/KatalonTestAnimales/proyecto.
```

```
prj" -retry=0 -retryFailedTestCases=false -testSuitePath="Test Suites/  
katalon" -executionProfile="default" -browserType="Chrome"  
-apiKey="87129c20-6518-4430-9564-1808aa69e734"
```

Se pegará en el campo Command arguments⁷, como se detalla en la figura 153.



Figura 153. Código Test de Katalon.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020) some recent research related to the topics covered in this work is exposed, as well as some tools that comply with the good development practices proposed by (Pastrana et al., 2019)

La ruta donde queda instalado Katalon en el servidor CentOS 7 es la siguiente: `/var/lib/jenkins/.katalon/6.3.3/Katalon_Studio_Engine_Linux_64-6.3.3`

Luego, en la sección *Acciones para ejecutar*, se añade la acción de *Publicar los resultados de Tests JUnit*. En el campo *Ficheros XML con los informes de tests*, se ingresa la ruta en donde se encuentran los archivos .xml, que están ubicados en este caso en la siguiente dirección: `KatalonTestAnimales/Reports/casoDeUsoCompleto/*/*.xml`⁸. Quedaría de la siguiente manera, ver figura 154:



Figura 154. Configuraciones JUnit.

Fuente: (Leiton Muñoz y Moya Loaiza, 2020)

⁷ Se recomienda que la versión a instalar sea inferior a la 7.0, ya que las versiones superiores presentan fallos de activación al ejecutar las pruebas en Jenkins, caso que no sucede en versiones inferiores.
⁸ Con los * se indica que ingrese a todos los archivos de la carpeta padre, con esto se verán todos los reportes que deje la ejecución del proyecto.

Proyectos realizados por Smart Campus

Para la fase I del proyecto Ecosistema Smart Campus se realizaron satisfactoriamente un total de 9 trabajos de grado, de los cuales el 56% (5 trabajos) se enfocaron en el eje de Smart Government y el restante 44% (4 trabajos) fueron de Smart People, como muestra la figura 155.

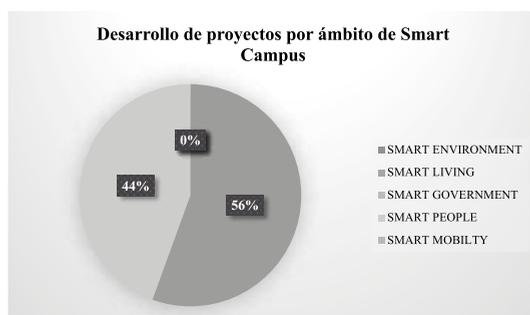


Figura 155. Porcentaje de proyectos ejecutados por eje Fase I.

Autora: Ana Milena Rojas Calero. Seguimiento de proyectos Smart Campus fase I

Así mismo, en la figura 156 se destaca la producción realizada por los investigadores del proyecto en dicha fase. Teniendo presente que el profesor Manuel Alejandro Pastrana participa con una codirección con el profesor Leandro Flórez Aristizábal.

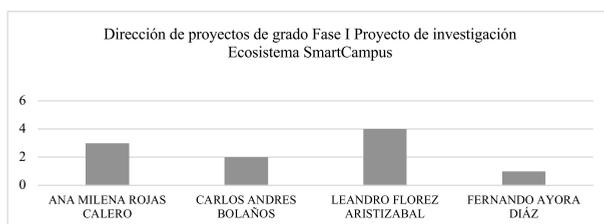


Figura 156. Proyectos dirigidos por investigador.

Autora: Ana Milena Rojas Calero. Seguimiento de proyectos Smart Campus fase I

A continuación, se da un detalle puntual sobre los proyectos realizados.

FASE I

Smart People - Ciudadanos

Trabajo de grado: Sistema de Información para plataforma IOS de eventos académicos y financieros de la Institución Universitaria Antonio José Camacho.

Estudiante(s): Cristian Fabián Barco Volveras

Director: Leandro Flórez Aristizábal

Datos de contacto director: learistizabal@admon.uniajc.edu.co

Periodo finalización: Segundo semestre del 2017

Programa: Ingeniería de Sistemas (Diurno)

Trabajo de grado: Sistema de Información para la integración de componentes de software que permita la implementación de un módulo que integre la funcionalidad de control de seguridad y consulta de notas para su acceso desde un aplicativo móvil dirigido a los estudiantes de la Institución Universitaria Antonio José Camacho

Estudiante(s): Fabián Esteban Capote Casas

Director: Leandro Flórez Aristizábal

Datos de contacto director: learistizabal@admon.uniajc.edu.co

Periodo finalización: Primer semestre del 2018

Programa: Ingeniería de Sistemas (Nocturno)

Trabajo de grado: Laboratorio Remoto de Redes

Estudiante(s): Lizeth Valentina Burbano Rodríguez-Jessica Yuliana Moreno Guerrero

Director: Ana Milena Rojas Calero

Datos de contacto director: amrojas@admon.uniajc.edu.co

Periodo finalización: Primer semestre del 2018

Programa: Ingeniería de Sistemas (Diurno)

Distinción: Trabajo de grado meritório

Trabajo de grado: Sistema para la Gestión de Comunicación de Carácter Académico entre los Estudiantes y Docentes en la Institución Universitaria Antonio José Camacho

Estudiante(s): Fernando Andrés Cifuentes Calderón - Nasly Katerine Escobar

Director: Leandro Flórez Aristizabal – Manuel Alejandro Pastrana Pardo

Datos de contacto director: learistizabal@admon.uniajc.edu.co

Periodo finalización: Primer semestre del 2018

Programa: Ingeniería de Sistemas (Diurno)

Distinción: Trabajo de grado meritório

Smart Government - Administración y gobierno

Trabajo de grado: Implementación de un sistema de Gestión Documental electrónico de archivo en la Institución Universitaria Antonio José Camacho

Estudiante(s): Paula Andrea Cañas Josa

Director: Ana Milena Rojas Calero

Datos de contacto director: amrojas@admon.uniajc.edu.co

Periodo finalización: Segundo semestre del 2018

Programa: Ingeniería de Sistemas (Diurno)

Distinción: Trabajo de grado meritório

Trabajo de grado: Desarrollo de Plataforma Web para la creación de formularios de encuestas para la Institución Universitaria Antonio José Camacho

Estudiante(s): Vanessa Rodríguez Galvis - Diego Armando Cortés Valencia

Director: Fernando Ayora Díaz

Datos de contacto director: fayora@admon.uniajc.edu.co

Periodo finalización: Segundo semestre del 2018

Programa: Tecnología en Sistemas (Nocturno)

Trabajo de grado: Sistema de información web para la gestión de contratos en la UNIAJC - Fase I

Estudiante(s): Juan Pablo Dorado Mesa - Alexander Castaño Ramírez

Director: Ana Milena Rojas Calero

Datos de contacto director: amrojas@admon.uniajc.edu.co

Periodo finalización: Segundo semestre del 2018

Programa: Ingeniería de Sistemas (Diurno)

Trabajo de grado: Software para la reserva de salas y equipos de cómputo para la Institución Universitaria Antonio José Camacho Fase I

Estudiante(s): Kevin Fabián Rodríguez Collazos - Héctor Fabio Rodríguez Gaviria

Director: Carlos Andrés Bolaños Ceballos

Datos de contacto director: cabo@admon.uniajc.edu.co

Periodo finalización: Segundo semestre del 2018

Programa: Ingeniería de Sistemas (Diurno)

Trabajo de grado: Sistema para la gestión de proyectos Smart Campus

Estudiante(s): Rubén Díaz Pulli - Manuela Cardona

Director: Ana Milena Rojas Calero

Datos de contacto director: amrojas@admon.uniajc.edu.co

Periodo finalización: Segundo semestre del 2018

Programa: Ingeniería de Sistemas (Diurno)

Distinción: Trabajo de grado meritório

FASE II

Para la fase II del proyecto Ecosistema Smart Campus se realizaron satisfactoriamente un total de 21 trabajos de grado, de los cuales el 43% (9 trabajos) se enfocaron en el eje de Smart Government, el 23% (5 trabajos) a Smart Environment, el 19% (4 trabajos) fueron de Smart People, el 9% a Smart Living (2 trabajos) y el restante 5% (1 trabajo) a Smart Mobility, evidenciando un crecimiento de alcance en los ejes trabajados dentro del macro proyecto, como muestra la figura 157.

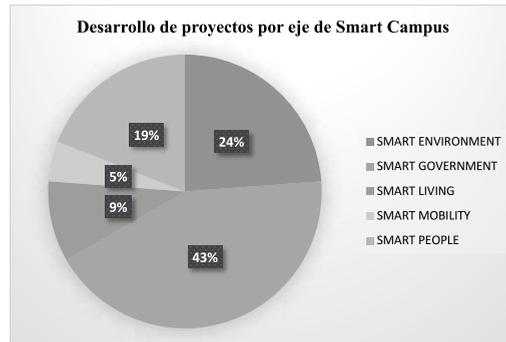


Figura 157. Porcentaje de proyectos ejecutados por eje Fase I.

Autora: Ana Milena Rojas Calero. Seguimiento de proyectos Smart Campus fase II

Así mismo, en la figura 158 se destaca la producción realizada por los investigadores del proyecto en dicha fase.

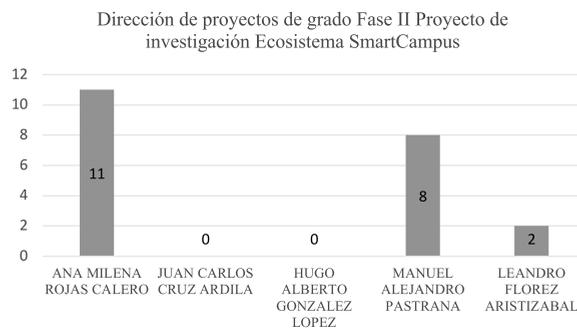


Figura 158. Proyectos dirigidos por investigador.

Autora: Ana Milena Rojas Calero. Seguimiento de proyectos Smart Campus fase II

A continuación, se da un detalle puntual sobre los proyectos realizados.

Smart People - Ciudadanos

Trabajo de grado: Aplicación que permite consultar notas de materias para estudiantes de la UNIAJC. (Actualización Android)

Estudiante(s): Fabio Capote

Director: Leandro Flórez Aristizábal

Datos de contacto director: learistizabal@admon.uniajc.edu.co

Periodo finalización: Programa: Ingeniería de Sistemas (Nocturno)

Trabajo de grado: Diseño y desarrollo de un sistema de información sobre dispositivos móviles para diagnósticos y quices en clases

Estudiante(s): Jorge Alexander Ortiz Cortés

Director: Leandro Flórez Aristizábal

Datos de contacto director: learistizabal@admon.uniajc.edu.co

Periodo finalización: Programa: Ingeniería de Sistemas (Nocturno)

Trabajo de grado: Sistema para la Gestión de Eventos Institucionales

Estudiante(s): Jaime Esteban Marulanda Vélez - Sebastián Ramírez Grisales

Director: Manuel Alejandro Pastrana Pardo

Datos de contacto director: mapastrana@admon.uniajc.edu.co

Periodo finalización: Primer semestre del 2020

Programa: Ingeniería de Sistemas (Nocturno)

Distinción: Trabajo de grado meritorio

Trabajo de grado: Actualización del software para la expedición de certificados y constancias dentro de la Institución Universitaria Antonio José Camacho

Estudiante(s): Raúl Andrés Hernández Ocampo - Stiven Castro Arias

Director: Manuel Alejandro Pastrana Pardo

Datos de contacto director: mapastrana@admon.uniajc.edu.co

Periodo finalización: Segundo semestre del 2020

Programa: Ingeniería de Sistemas (Nocturno)

Smart Government – Administración y gobierno

Trabajo de grado: Desarrollo de aplicación móvil de consulta y módulo de integración con el sistema „Academusoft“ para el software de reserva de salas y equipos para la „UNIAJC“ Fase II

Estudiante(s): Kevin Fabián Rodríguez Collazos - Héctor Fabio González Gaviria

Directores: Carlos Andrés Bolaños Ceballos- Ana Milena Rojas Calero

Datos de contacto directores: cabo@admon.uniajc.edu.co amrojas@admon.uniajc.edu.co

Periodo finalización: Programa: Ingeniería de Sistemas (Nocturno)

Trabajo de grado: Implementación de un dispositivo de acceso inteligente mediante tarjetas RFID y sensor biométrico para huella dactilar en la Institución Universitaria Antonio José Camacho

Estudiante(s): Luis Eduardo Hurtado Grijalba

Director: Ana Milena Rojas Calero

Datos de contacto director: amrojas@admon.uniajc.edu.co

Periodo finalización: Programa: Ingeniería Electrónica (Nocturno)

Distinción: Trabajo de grado meritorio

Trabajo de grado: Aplicación web para gestión del ascenso del escalafón de Docentes de la UNIAJC

Estudiante(s): Cristian Gómez - Erick Pulido Neira

Director: Carlos Andrés Bolaños Ceballos- Ana Milena Rojas Calero

Datos de contacto director: cabo@admon.uniajc.edu.co amrojas@admon.uniajc.edu.co

Periodo finalización: Programa: Ingeniería de Sistemas (Nocturno)

Trabajo de grado: Diseño e implementación de solución analítica B.I. para la Gestión de Indicadores Académicos de la Institución Universitaria Antonio José Camacho.

Estudiante(s): Jhoan Sebastián Giraldo Mosquera - Juan David Ortiz Guevara

Director: Ana Milena Rojas Calero - Carlos Andrés Bolaños Ceballos

Datos de contacto director: amrojas@admon.uniajc.edu.co cabo@admon.uniajc.edu.co

Periodo finalización: Segundo semestre del 2020

Programa: Ingeniería de Sistemas (Nocturno)

Distinción: Trabajo de grado meritorio

Trabajo de grado: Sistema para la gestión de solicitudes a la Facultad de Ingenierías de la Institución Universitaria Antonio José Camacho

Estudiante(s): Diego Andrés Gutiérrez Díaz - Gilberto Delgado Velasco

Director: Manuel Alejandro Pastrana Pardo

Datos de contacto director: mapastrana@admon.uniajc.edu.co

Periodo finalización: Primer Semestre 2020

Programa: Ingeniería de Sistemas (Nocturno)

Distinción: Trabajo de grado meritorio

Trabajo de grado: Sistema de gestión del índice de deserción en la UNIAJC.

Estudiante(s): María Camila Silva Moreno

Director: Manuel Alejandro Pastrana Pardo

Datos de contacto director: mapastrana@admon.uniajc.edu.co

Periodo finalización: pendiente por entregar

Programa: Ingeniería de Sistemas (Nocturno)

Trabajo de grado: Adaptación de herramienta para visualizar el control y monitoreo de los recursos de la red de la UNIAJC

Estudiante(s): Gustavo Borja

Director: Ana Milena Rojas Calero

Datos de contacto director: amrojas@admon.uniajc.edu.co

Periodo finalización: Pendiente por entregar

Programa: Ingeniería de Sistemas (Diurno)

Smart Enviroment - Entorno y eficiencia

Trabajo de grado: Implementación de un dispositivo de monitoreo para un sistema inteligente con sensores distribuidos para las variables de temperatura y humedad el NOC de la UNIAJC

Estudiante(s): Joseph Caicedo Ambuila - Andrés Mauricio Rodríguez Bermúdez - Juan Felipe Collazos Prieto

Director: Ana Milena Rojas - Mario German Domínguez

Datos de contacto director: amrojas@admon.uniajc.edu.co

Periodo finalización: Pendiente por entregar

Programa: Tecnología en Electrónica Industrial

Trabajo de grado: Desarrollo de la aplicación móvil para la Unidad Municipal de Asistencia Técnica Agropecuaria (UMATA) en el Municipio de Versalles-Valle

Estudiante(s): Carlos Alberto Valencia Cocomá - Ronal Andrés Tamayo Zapata

Director: Manuel Alejandro Pastrana Pardo

Datos de contacto director: mapastrana@admon.uniajc.edu.co

Periodo finalización: Primer semestre del 2020

Programa: Ingeniería de Sistemas (Nocturno)

Trabajo de grado: Documentación e implementación de buenas prácticas DevOps con herramientas de automatización de pruebas y de despliegue a través del refinamiento de software de Smart Campus

Estudiante(s): William Andrés Leiton Muñoz - Brayan Alexis Moya Loaiza

Director: Manuel Alejandro Pastrana Pardo

Datos de contacto director: mapastrana@admon.uniajc.edu.co

Periodo finalización: Segundo semestre del 2017

Programa: Ingeniería de Sistemas (Diurno)

Distinción: Trabajo de grado meritorio

Trabajo de grado: Diseño de un dispositivo de medición inteligente para la monitorización del consumo energético en una sala de sistema de la UNIAJC

Estudiante(s): Diego Alejandro López García - Jhonatan Jair Mina Carabalí - Sergio Manolo Ordóñez

Director: Ana Milena Rojas Calero

Datos de contacto director: amrojas@admon.uniajc.edu.co

Periodo finalización: Pendiente por entregar

Programa: Ingeniería Electrónica (Diurno)

Trabajo de grado: Prototipo de un punto ecológico inteligente para fomentar la cultura ambiental en las instalaciones de la UNIAJC

Estudiante(s): Andrés Camilo Angulo Correa

Director: Ana Milena Rojas Calero – Erika Sarria

Datos de contacto director: amrojas@admon.uniajc.edu.co esarria@admon.uniajc.edu.co

Periodo finalización: Segundo semestre 2008

Programa: Ingeniería Electrónica (Diurno)

Smart Mobility - Movilidad

Trabajo de grado: Diseño e implementación de la APP Green Car UNIAJC, como estrategia de movilidad inteligente dirigida a la comunidad académica de la UNIAJC

Estudiante(s): Luis Alberto Mora Montoya - José Roberto Granada Chacón

Director: Manuel Alejandro Pastrana Pardo

Datos de contacto director: mapastrana@admon.uniajc.edu.co

Periodo finalización: Para sustentar 2021 primer semestre

Programa: Ingeniería de Sistemas (Nocturno)

Distinción:

Smart Living - Estilo de vida

Trabajo de grado: Aplicación Móvil Cafetería Virtual UNIAJC

Estudiante(s): Nicolás Ortiz - Nicolás Bermúdez

Director: Manuel Alejandro Pastrana Pardo

Datos de contacto director: mapastrana@admon.uniajc.edu.co

Periodo finalización: Primer periodo 2020

Programa: Ingeniería de Sistemas (Nocturno)

Distinción: Ninguna

Trabajo de grado: Desarrollo de un sistema de información web para la gestión de historias clínicas realizadas por el programa de salud de Bienestar Universitario de la UNIAJC

Estudiante(s): Lina Vanessa López Castañeda

Director: Ana Milena Rojas Calero

Datos de contacto director: amrojas@admon.uniajc.edu.co

Periodo finalización:

Programa: Ingeniería de Sistemas (Nocturno)

Distinción: Ninguna

Publicaciones en las que participó el equipo de investigadores

A continuación, se hace mención a las publicaciones obtenidas por los investigadores del proyecto:

Producción bibliográfica - Artículo - Publicado en revista especializada

Autores: Fabián Esteban Capote, Leandro Floréz Aristizábal, Ana Milena Rojas Calero, Carlos Andrés Bolaños Ceballos, Sandra Cano, César A. Collazos.

Título: "Development of a SOA Platform Support the Integration of Software Components Based On Mobile Devices for a Smart Campus".

En: Cali, Colombia.

Advances in Computing. CCC 2017. Communications in Computer and Information Science, vol. 735. Springer, Cham. https://doi.org/10.1007/978-3-319-66562-7_48

Palabras clave: Smart Campus Service-Oriented, Architecture Android, Web service Database Academusoft

Producción bibliográfica - Artículo - Publicado en revista especializada

Autores: Hugo Ordonez Eraso, Manuel Alejandro Pastrana Pardo, Carlos Alberto Cobos Lozada.

Título: "ISO 29110 en Colombia: de la teoría a la práctica".

En: Colombia

Revista: Guillermo De Ockham ISSN: 1794-192X ed: UNIVERSIDAD DE SAN BUENAVENTURA v.18 fasc.1 p.1 - 15, 2020, DOI: 10.21500/22563202.4299

Palabras clave: ISO 29110, buenas prácticas, Desarrollo de software,

Producción bibliográfica - Artículo - Publicado en revista especializada

Autores: Manuel Alejandro Pastrana Pardo, Hugo Ordonez Eraso, Ana Milena Rojas Calero.

Título: "Ensuring Compliance with Sprint Requirements in SCRUM".

En: Singapur Advances in Intelligent Systems and Computing ISSN: 2194-5357
ed: Springer Science + Business Media v.924 fasc. p.33 - 45, 2019, DOI:
10.1007/978-981-13-6861-5_3

Palabras clave: Scrum, Procesos Para Desarrollo de Software, Software Quality Assurance-SQA.

Producción bibliográfica - Artículo - Publicado en revista especializada

Autores: Manuel Alejandro Pastrana Pardo, Hugo Ordonez Eraso, Fernando Andrés Cifuentes Calderón.

Título: Arquitectura de Integración para Servicios y Soluciones Smart Campus

Revista: INGCUC ISSN: 0122-6517 ed: IngCUC 16. fasc.2, 2020, DOI:
10.17981/ingecuc.16.2.2020.21

Palabras clave: ESB, Arquitectura de Integración, Arquitectura orientada a servicios, SOA.

Producción bibliográfica - Artículo - Publicado en revista especializada

Autores: Andrés Camilo Angulo Correa, Erika Sarria Navarro

Tipo: Capítulo de libro.

Título: Prototipo de un punto ecológico inteligente para fomentar la cultura ambiental en las instalaciones de la UNIAJC Cali, Colombia, septiembre, 2018.

En: La Investigación Formativa, ISBN:978-958-8292-84-7, pp. 222 - 226, 2020

Palabras clave: Reciclaje Inteligente, Reconocimiento Automático de Voz

Capítulos de libro

Producción bibliográfica - tipo capítulo de libro

Autores: Manuel Alejandro Pastrana Pardo, Hugo Ordonez Eraso.

Tipo: Capítulo de libro.

Título: "A model approach to ISO-29110 for very small enterprises".

Proceedings of the XXII Iberoamerican Conference on Software Engineering, ClbSE 2019, La Habana, Cuba, April 22-26, 2019. Curran Associates 2019.

En: Colombia. ISBN: 978-1-5108-8795-4 ed: Ibero-American Conference on Software Engineering, v, pp. 517-530.

Producción bibliográfica - tipo capítulo de libro

Autores: Manuel Alejandro Pastrana Pardo, Hugo Ordonez Eraso.

Tipo: Capítulo de libro.

Título: "La Elicitación de Requisitos mediante la técnica Inception Deck en las

empresas de Santiago de Cali”.

Proceedings of XXIV Congreso Colombiano de Computación, 14CCC, Pasto, Colombia, septiembre 25-27, 2019.

En: Colombia. ISBN: pendiente. Ed.: UNICESMAG, 2020

Producción bibliográfica - tipo capítulo de libro

Autores: Manuel Alejandro Pastrana Pardo, Hugo Ordonez Erasó.

Tipo: Capítulo de libro.

Título: “Inception Deck un camino hacia el desarrollo de Sof Skills requeridas por el sector TI en Colombia”.

Proceedings of XXIV Congreso Colombiano de Computación, 14CCC, Pasto, Colombia, septiembre 25-27, 2019.

En: Colombia ISBN: pendiente. Ed.: UNICESMAG, 2020

Producción bibliográfica - Otro artículo publicado - Revista de divulgación

Autores: Ana Milena Rojas Calero, Manuel Alejandro Pastrana Pardo.

Título: “Desarrollo de un sistema de información para la comercialización de cupones de alta segmentación en facturas de servicios públicos”.

En: Colombia. 2018. Actitud. ISSN: 1909-8510 pp.5-14 v.15

Palabras clave: Desarrollo de software, Arquitectura de software,

Áreas: Ingeniería y Tecnología, Ingenierías Eléctrica, Electrónica e Informática, Ingeniería de Sistemas y Comunicaciones.

Producción bibliográfica - Otro artículo publicado - Periódico de noticias

Autores: Ana Milena Rojas Calero, Manuel Alejandro Pastrana Pardo.

Título: “Patrones de diseño de software para modelos arquitecturales en smart campus basado en inception deck e ingeniería kaisen”.

En: Colombia. 2018. Sapiencia. ISSN: 0286-2204 pp.52-64 v.12

Palabras clave: Arquitectura de software, Desarrollo de software, smart campus, ingeniería kaisen,

Áreas: Ingeniería y Tecnología, Ingenierías Eléctrica, Electrónica e Informática, Ingeniería de Sistemas y Comunicaciones.

Producción bibliográfica - Otro artículo publicado - Periódico de noticias

Autores: Manuel Alejandro Pastrana Pardo, Ana Milena Rojas Calero.

Título: “Smart Living: cafetería virtual para la institución universitaria Antonio José Camacho”.

En: Colombia. 2018. Sapiencia. ISSN: 0286-2204, pp.52-64 v.2

Palabras: Smart Campus, Smart Living, Desarrollo web, arquitectura de software

Áreas: Ingeniería y Tecnología, Ingenierías Eléctrica, Electrónica e Informática, Ingeniería de Sistemas y Comunicaciones.

Eventos científicos en los que participo el equipo de investigadores Registros de software

Nombre del evento: 12 Congreso Colombiano de Computación

Tipo de evento: Congreso

Ámbito: Internacional

Realizado el: 2017-09-20, 2019-09-22 en CALI - Universidad Autónoma de Colombia.

Productos asociados

- **Nombre del producto:** Development of a SOA Platform Support the Integration of Software Components Based On Mobile Devices for a Smart Campus
- **Tipo de producto:** Producción técnica - Presentación de trabajo - Ponencia

Participantes

- **Nombre:** FABIÁN ESTEBAN CAPOTE CASAS
- **Rol en el evento:** Ponente

Nombre del evento: 14 Congreso Colombiano de Computación

Tipo de evento: Congreso

Ámbito: Internacional

Realizado el: 2019-09-25, 2019-09-27 en PASTO - Universidad Mariana

Productos asociados

- **Nombre del producto:** Inception Deck un camino hacia el desarrollo de Softskills para el sector TI Colombia
- **Tipo de producto:** Producción técnica - Presentación de trabajo - Ponencia
- **Nombre del producto:** La Elicitación de Requisitos mediante la técnica Inception Deck en las empresas de Santiago de Cali

- **Tipo de producto:** Producción técnica - Presentación de trabajo - Ponencia

Participantes

- **Nombre:** MANUEL ALEJANDRO PASTRANA PARDO
- **Rol en el evento:** Ponente

Nombre del evento: IX ENCUENTRO SOFTHARD 2019

Tipo de evento: Encuentro

Ámbito: Nacional

Realizado el: 2019-05-02, 2019-05-02 en CALI - Institución Universitaria Antonio José Camacho

Participantes

- **Nombre:** MANUEL ALEJANDRO PASTRANA PARDO
- **Rol en el evento:** Ponente

Nombre del evento: IX TECNOTIC 2019

Tipo de evento: Encuentro

Ámbito: Nacional **Realizado el:** 2019-09-26, 2019-09-26 en CALI - Institución Universitaria Antonio José Camacho

Productos asociados

- **Nombre del producto:** Desarrollo de una aplicación (UMATA) multiplataforma en Ionic basado en microservicios.
- **Tipo de producto:** Producción técnica - Presentación de trabajo – Ponencia

Participantes

- **Nombre:** MANUEL ALEJANDRO PASTRANA PARDO, **Rol en el evento:** Ponente
- **Nombre:** RONAL ANDRES TAMAYO ZAPATA,
- **Rol en el evento:** Ponente
- **Nombre:** CARLOS ALBERTO VALENCIA COCOMA
- **Rol en el evento:** Ponente

Nombre del evento: IC4S2018- INTERNATIONAL CONFERENCE ON COMPUTER, COMMUNICATION AND COMPUTATIONAL SCIENCES

Tipo de evento: Otro

Ámbito: Internacional

Realizado el: 2018-10-20, 2018-10-21 en Bangkok - MANDARIN HOTEL BANGKOK, BANGKOK, THAILAND

Productos asociados

- **Nombre del producto:** Ensuring compliance with sprint requirements in SCRUM Preventive quality assurance in SCRUM
- **Tipo de producto:** Producción técnica - Presentación de trabajo - Ponencia

Participantes

- **Nombre:** MANUEL ALEJANDRO PASTRANA PARDO Rol en el evento: Ponente

Nombre del evento: ENCUENTRO CIENTÍFICO Y EDUCATIVO DE EXPERIENCIAS INVESTIGATIVAS

Tipo de evento: Encuentro

Ámbito: Nacional

Realizado el: 2016-09-29, 2016-10-01 en CALL - Institución Universitaria Antonio José Camacho

Productos asociados

- **Nombre del producto:** Ecosistema Smart Campus UNIAJC.
- **Tipo de producto:** Producción técnica - Presentación de trabajo - Ponencia

Participantes

- **Nombre:** ANA MILENA ROJAS CALERO
- **Rol en el evento:** Ponente

Nombre del evento: 2do ENCUENTRO CIENTÍFICO Y EDUCATIVO DE EXPERIENCIAS INVESTIGATIVAS

Tipo de evento: Encuentro

Ámbito: Nacional

Realizado el: 2017-10-07, 2017-10-08 en CALI - Institución Universitaria Antonio José Camacho

Productos asociados

- **Nombre del producto:** Prototipo de Laboratorio Remoto de Redes.
- **Tipo de producto:** Producción técnica - Presentación de trabajo - Ponencia

Participantes

- **Nombre:** YESICA YULIANA MORENO GUERRERO
- **Rol en el evento:** Ponente
- **Nombre:** LISETH VALENTINA BURBANO RODRÍGUEZ
- **Rol en el evento:** Ponente
- **Nombre:** ANA MILENA ROJAS CALERO
- **Rol en el evento:** Ponente

Nombre del evento: 2do ENCUENTRO CIENTÍFICO Y EDUCATIVO DE EXPERIENCIAS INVESTIGATIVAS

Tipo de evento: Encuentro

Ámbito: Nacional

Realizado el: 2017-10-07, 2017-10-08 en CALI - Institución Universitaria Antonio José Camacho

Productos asociados

- **Nombre del producto:** Prototipo de Laboratorio Remoto de Redes.
- **Tipo de producto:** Producción técnica - Presentación de trabajo - Ponencia

Participantes

- **Nombre:** YESICA YULIANA MORENO GUERRERO
- **Rol en el evento:** Ponente

- **Nombre:** LISETH VALENTINA BURBANO RODRÍGUEZ

- **Rol en el evento:** Ponente

- **Nombre:** ANA MILENA ROJAS CALERO

- **Rol en el evento:** Ponente

Nombre del evento: ece2i ENCUENTRO CIENTÍFICO Y EDUCATIVO DE EXPERIENCIAS INVESTIGATIVAS

Tipo de evento: Encuentro

Ámbito: Nacional

Realizado el: 01-8-09-16, en CALL - Institución Universitaria Antonio José Camacho.

Productos asociados

- **Nombre del producto:** Punto Ecológico Inteligente para fomentar la cultura ambiental en las instalaciones de la Institución Universitaria Antonio José Camacho

- **Tipo de producto:** Producción técnica - Presentación de trabajo - Ponencia

Participantes

- **Nombre:** ANDRÉS CAMILO ANGULO CORREA.

- **Rol en el evento:** Ponente

- **Nombre:** ERIKA SARRIA

- **Rol en el evento:** Ponente

Nombre del evento: V ENCUENTRO REGIONAL DE SEMILLEROS DE INVESTIGACIÓN IES TÉCNICAS Y TECNOLÓGICA DEL VALLE DEL CAUCA

II ENCUENTRO INTERNACIONAL DE GRUPOS Y SEMILLEROS DE INVESTIGACIÓN TÉCNICA PROFESIONAL Y TECNOLÓGICA

Tipo de evento: Encuentro

Ámbito: Nacional

Realizado el: 2017-04-26, 2017-04-27 en SANTANDER DE QUILICHAO CAUCA - Corporación Universitaria COMFACAUCA.

Productos asociados

- **Nombre del producto:** Sistema de información para la integración de componentes de software que permita la implementación de un módulo que integre la funcionalidad de control de seguridad y consulta de notas para su acceso desde un aplicativo móvil dirigido a los estudiantes de la Institución Universitaria Antonio José Camacho.
- **Tipo de producto:** Producción técnica - Presentación de trabajo - Ponencia

Participantes

- **Nombre:** FABIÁN ESTEBAN CAPOTE CASAS
- **Rol en el evento:** Ponente
- **Nombre:** LEANDRO FLÓREZ ARISTIZÁBAL
- **Rol en el evento:** Ponente

Nombre del evento: VI ENCUENTRO REGIONAL DE SEMILLEROS DE INVESTIGACIÓN IES TÉCNICAS Y TECNOLÓGICA DEL VALLE DEL CAUCA
II ENCUENTRO INTERNACIONAL DE GRUPOS Y SEMILLEROS DE INVESTIGACIÓN TÉCNICA PROFESIONAL Y TECNOLÓGICA

Tipo de evento: Encuentro

Ámbito: Nacional

Realizado el: 018-04-25, 2017-04-26 en CALI - UAOTEC, Sede San Fernando

Productos asociados

- **Nombre del producto:** Prototipo de Laboratorio Remoto de Redes
- **Tipo de producto:** Producción técnica - Presentación de trabajo - Ponencia

Participantes

- **Nombre:** LISETH VALENTINA BURBANO RODRÍGUEZ
- **Rol en el evento:** Ponente
- **Nombre:** YESICA YULIANA MORENO GUERRERO
- **Rol en el evento:** Ponente

- Nombre: ANA MILENA ROJAS CALERO
- Rol en el evento: Ponente

Registros de software

Producción técnica - Softwares - Computacional

Manuel Alejandro Pastrana Pardo. Sistema de registro y reportes de las acciones de proyección social de la Institución Universitaria Antonio José Camacho de la ciudad de Cali.

Nombre comercial: Sistema de registro y reportes de las acciones de proyección social de la Institución Universitaria Antonio José Camacho de la ciudad de Cali.

contrato/registro: En: Colombia, 2018

Plataforma: El usuario puede tener acceso al aplicativo mediante un dispositivo que tenga un web browser y esté conectado a una red o intranet, por medio del puerto 8080 (por defecto) usando el protocolo TCP/IP a través de HTTP. Se establece un puente de conexión entre el usuario y el dispositivo, y el servidor de aplicaciones glassfish que contiene la aplicación Java desplegada. El servidor de aplicaciones, mediante el uso de un pool de conexiones y sesiones, se conecta a un servidor de base de datos Oracle a través del puerto 1521 (por defecto) usando el protocolo TCP/IP con un usuario y contraseña de conexión, de donde extrae toda la data permitida y solicitada por el usuario mediante el web browser.

Palabras clave: Arquitectura de software, Procesos Para Desarrollo de Software, Scrum, Spring Framework.

Áreas: Ingeniería y Tecnología, Ingenierías Eléctrica, Electrónica e Informática, Ingeniería de Sistemas y Comunicaciones.

Producción técnica - Softwares - Computacional

Manuel Alejandro Pastrana Pardo, Leandro Flórez Aristizábal. Sistema para la Gestión de Comunicación de Carácter Académico entre los Estudiantes y Docentes en la Institución Universitaria Antonio José Camacho.

Nombre comercial: Sistema para la Gestión de Comunicación de Carácter Académico entre los Estudiantes y Docentes en la Institución Universitaria Antonio José Camacho.

Contrato/registro: En: 2018.

Plataforma: El aplicativo funciona para las plataformas Android e iOS.

Palabras: Arquitectura de software, Desarrollo de software, Procesos Para Desarrollo de Software, Scrum, smart campus.

Áreas: Ingeniería y Tecnología, Ingenierías Eléctrica, Electrónica e Informática, Ingeniería de Sistemas y Comunicaciones.

Producción técnica - Softwares - Computacional

Ana Milena Rojas Calero, Juan Pablo Dorado Mesa. Desarrollo de una plataforma Web para la Gestión de Contratos de Mínima Cuantía y Contratación Directa en la Institución Universitaria Antonio José Camacho.

Nombre comercial: Desarrollo de una plataforma Web para la Gestión de Contratos de Mínima Cuantía y Contratación Directa en la Institución Universitaria Antonio José Camacho.

Contrato/registro: En: 2018.

Plataforma: Unos de los aspectos más relevantes en el cumplimiento de la función pública es el manejo de la gestión contractual, por la incidencia directa que esta tiene sobre el cumplimiento de las funciones constitucionales, que son asignadas en el cumplimiento de las funciones que les son asignadas en la inversión y en el manejo del presupuesto público.

El aplicativo está en desarrollado en PHP, la plataforma web permite elaborar los estudios previos para la contratación directa y de mínima cuantía; también permite llevar un control documental de las actas o documentos que son elaboradas por el interventor del contrato, posterior a la firma de la suscripción del mismo.

Palabras clave: Contratación pública, estudio previo, Smart Government, procesos de desarrollo de software, Scrum, smart campus.

Áreas: Ingeniería y Tecnología, Ingenierías Eléctrica, Electrónica e Informática, Ingeniería de Sistemas y Comunicaciones.

Producción técnica - Softwares - Computacional

Ana Milena Rojas Calero, Manuela Cardona Arias, Rubens Diez Puli. Desarrollo de un Sistema de Información para la gestión de proyectos de Smart Campus

Nombre comercial: Sistema de Información para la gestión de proyectos de Smart Campus

Contrato/registro: En: 2019.

Plataforma: El sistema de Información para la gestión de proyectos de Smart Campus, permite ver en tiempo real la visibilidad del flujo de trabajo de los equipos de desarrollo que se encuentran vinculados al proyecto de investigación. El tablero consta de cuatro columnas (Por hacer, Haciendo, Pruebas, Terminado). Este proyecto busca contribuir significativamente a mejorar el control y seguimiento de los proyectos de grado en modalidad asistida en el grupo de investigación GRINTIC, facilitando al director de proyecto y estudiantes, la comunicación, el seguimiento de tareas, la gestión documental y finalmente la gestión del conocimiento de cada proyecto. Es importante resaltar que el marco de trabajo con el cual se realiza el proceso de elaboración del proyecto está basado en el framework Scrum, su desarrollo se centra en la aplicación de tableros utilizando la metodología Kanban.

Palabras clave: Smart Campus, Gestión de proyectos, Kanban, tablero, gestión de conocimiento, gestión de documentos

Áreas: Ingeniería y Tecnología, Ingenierías Eléctrica, Electrónica e Informática, Ingeniería de Sistemas y Comunicaciones.

Impacto en la comunidad

Los impactos alcanzados en la fase I y fase II del proyecto de investigación Ecosistema Smart Campus UNIAJC serán abordados desde dos perspectivas: funciones sustantivas y funciones adjetivas de las Instituciones de Educación Superior (IES).

La primera tiene que ver con las funciones misionales de la educación, enunciadas en el marco de la Ley General de Educación 115 del año 1994, que son: Docencia, Investigación y Extensión. Esta última entendida en un sentido mucho más amplio en las IES como la proyección social. No obstante, en el mundo hay una fuerte tendencia que ha sido acogida en el Plan Estratégico de Desarrollo de la Institución Universitaria Antonio José Camacho 2020-2030, para que el Bienestar Universitario se reconozca también como una función sustantiva y, en tal sentido, se hará el análisis de los impactos misionales para estos cuatro ejes de desarrollo.

De otro lado, la segunda perspectiva desde la cual se analizarán los impactos es desde el punto de vista de las funciones adjetivas, que son funciones que no hacen parte del núcleo misional del desarrollo, pero que comprenden las acciones y programas asociados a mejorar la calidad y eficiencia de las funciones misionales. En tal sentido, son todas las áreas de la universidad que trabajan para que las funciones misionales puedan desarrollarse de la forma más adecuada. Hacen parte de las funciones adjetivas: las funciones administrativas, el mercadeo, la gestión tecnológica y la gestión logística. Es por esa razón que también se analizarán los impactos que ha tenido el proyecto en estas áreas de desempeño de la universidad.

Si bien, históricamente los pilares fundamentales del desarrollo de las funciones misionales en el mundo han sido la docencia y la investigación, en la universidad latinoamericana, en términos generales, estos dos pilares no han sido fáciles de conciliar, dado que son producto de modelos universitarios primigenios clásicos, con un fuerte arraigo euro centrista, en los cuales se privilegiaban con diferentes pesos cada una de estas funciones misionales.

Este fenómeno ha hecho que en las universidades se genere una tensión entre las funciones de docencia y las funciones de investigación. En este escenario, los profesores reclaman más tiempo para poder desarrollar la investigación, lo que termina generando una lucha entre el tiempo del docente para ejercer las funciones de investigación y el tiempo destinado a las funciones de docencia.

Uno de los principales impactos que ha tenido el proyecto ha sido propiciar un escenario de diálogo entre estas dos funciones misionales. En consecuencia, el desarrollo de esta iniciativa ha permitido la articulación curricular de las estructuras investigativas con el desarrollo de la oferta académica, que se realiza a través de los docentes investigadores. Esta estrategia nutre el proceso de construcción de conocimiento a través de la vinculación de estudiantes de los diferentes cursos en los procesos de investigación, permitiendo que los investigadores del grupo GRINTIC puedan gestionar el talento humano en las asignaturas que estratégicamente se han escogido para apoyar el desarrollo del proyecto de investigación. Estas asignaturas son: programación, ingeniería de software, programación de dispositivos móviles, temática de grado, proyecto integrador. De esta forma, los profesores investigadores han tenido la oportunidad de poder identificar los estudiantes que harán parte del colectivo de investigación, los cuales podrán vincularse al proyecto mediante el ingreso al semillero de investigación ITMedia y a través de su trabajo en

estos colectivos; estos estudiantes pueden iniciar el desarrollo de proyectos de grado en modalidad asistida en el grupo de investigación.

Al igual que la estructura curricular ha provisto talento humano para el desarrollo del proyecto de investigación, de forma recíproca, la estructura investigativa ha contribuido con el proceso de actualización curricular mediante el desarrollo de contenidos pertinentes y acordes a las necesidades de la industria del software, experiencias educativas basadas en el aprendizaje significativo, traídas de los resultados de investigación y plataformas tecnológicas modernas para apoyar el desarrollo de los procesos de formación.

Otro impacto en la función misional de la investigación está relacionado con el fortalecimiento de los indicadores y características de la acreditación de alta calidad. En particular, con los referidos a la investigación, innovación, creación artística y cultural, así como también al de visibilidad nacional e internacional. Es importante mencionar que, a través de los diferentes resultados de investigación, se aportaron productos para la categorización del grupo de investigación GRINTIC en las convocatorias de medición de grupos, desarrollo tecnológico o de innovación y el reconocimiento de investigadores del Sistema Nacional de Ciencia y Tecnología e Innovación (SNCTEI), definidos por Colciencias. Aquí, efectivamente resalta como aporte significativo el desarrollo de productos de nuevo conocimiento, productos de apropiación social de conocimiento y productos de actividades relacionadas con la formación del talento humano, entre los que podemos mencionar: la publicación de artículos en revistas indexadas, la participación en encuentros científicos a nivel nacional e internacional, encuentros de semilleros, consecución de registros de software, desarrollo de talleres, dirección de trabajos de grado, los cuales se encuentran expuestos en el capítulo de resultados de este libro.

Estos resultados han favorecido no sólo los procesos de medición del grupo, sino que también han permitido a los docentes avanzar en los procesos de desarrollo y cualificación profesoral propios de la institución, participar en eventos nacionales e internacionales de divulgación de conocimiento, publicación de resultados de investigación, registros de software y demás actividades derivadas de los procesos de investigación, financiadas por la Institución y que contribuyen al mejoramiento del escalafón de los docentes y al ascenso del perfil como investigador en Colciencias.

Los estudiantes investigadores también son beneficiarios de algunas de estas medidas de fomento de la investigación, mediante la financiación de la movilidad a eventos de divulgación, encuentros de semilleros y publicación de sus resultados en revistas científicas o de divulgación.

Desde el punto de vista de la función de proyección social también se ha conseguido un impacto muy relevante, a través del desarrollo del proyecto integrado con las funciones de investigación por medio del desarrollo de proyectos de grado que están orientados a la solución de problemas organizacionales del sector educativo desde un enfoque metodológico de marcos de trabajo ágil como: SCRUM, prácticas DevOps para el aseguramiento de la calidad del software y la adopción de marcos internacionales para la gestión de proyectos, lo que ha permitido a nuestros estudiantes hacer el cierre de brechas entre los conocimientos académicos de los procesos de formación universitaria y las necesidades reales de la industria del software.

La atención de estas necesidades a través de procesos empresariales ha permitido que el estudiante se acerque a una experiencia significativa en términos de su desarrollo profesional, potenciando sus oportunidades de inserción laboral, de mejoramiento de su calidad de vida y de su desarrollo profesional, convirtiendo al semillero de investigación en un referente para los centros tecnológicos de mayor importancia ubicados en el Valle del Cauca, Medellín y Bogotá, a los cuales han llegado varios de nuestros estudiantes.

También en el eje de proyección social y con la colaboración de profesores del equipo de investigadores, se realizó una consultoría a la fundación FUNIAJC en la creación del Centro de Formación de Capacidades Laborales y Empresariales, el diseño y actualización de ambientes de aprendizaje y el desarrollo curricular de una oferta académica especializada para el sector TIC.

Desde el punto de vista de la función misional del bienestar universitario, se han producido impactos positivos en varios niveles de la organización, el primero de ellos directamente en el equipo de jóvenes investigadores del semillero IT/Media, que por su participación en esta iniciativa son elegibles para el estímulo denominado becario de investigación, a través del cual el estudiante recibe un apoyo financiero equivalente al 80% de los derechos de matrícula y un auxilio de transporte que busca mitigar las condiciones de vulnerabilidad de los estudiantes de la Institución Universitaria Antonio José Camacho, los cuales en un 97% pertenecen a los estratos socioeconómicos 1, 2 y 3.

A nivel organizacional, también se tienen impactos significativos desde el eje misional del bienestar en la gestión de citas e historias clínicas para la orientación psicosocial, gestión de cafeterías en el campus universitario y en la generación de apoyos financieros mediante el programa referidos.

Por otra parte, están los impactos alcanzados desde las funciones adjetivas, en las cuales también hay resultados en diferentes aspectos. El primero de ellos tiene que ver con la gestión financiera, en tanto que se ha logrado el incremento de número de estudiantes resultado de la implementación de los proyectos: la integración del CRM con el sistema ERP empresarial Academusoft, la implementación de un Call Center y el desarrollo de la App de referidos Unidos Unicamacho, lo cual ha permitido generar estrategias de marketing digital y un permanente control de la gestión de cierre de venta de servicios educativos, mediado por el monitoreo del proceso de maduración de los aspirantes preinscritos a estudiantes matriculados, a través de sistemas de inteligencia de negocios. El incremento de estudiantes que contribuye a la ocupación de la capacidad instalada ociosa ha tenido un impacto en el mejoramiento de la eficiencia organizacional y ha favorecido el mejoramiento de la estabilidad y las condiciones de sostenibilidad de la universidad.

Dentro de las funciones adjetivas está también la gestión tecnológica, en la cual se ha avanzado en la estandarización del proceso de desarrollo de software mediante la implementación de buenas prácticas, el aseguramiento de calidad preventiva y el fortalecimiento de infraestructura tecnológica de servidores, consolidando una plataforma de desarrollo que ha hecho que el espacio dedicado para el semillero ITMedia se convierta en un laboratorio de práctica para los jóvenes investigadores que con las soluciones desarrolladas en modalidad de proyectos de grado contribuyen a la transformación digital de la universidad.

En el ámbito académico-administrativo, los desarrollos que han tenido mayor impacto han sido el desarrollo de la App de consulta de notas, la App de certificados académicos, la App para la comunicación docente-estudiante, la App calendario académico, la implementación de la ventanilla única, la App para la elaboración de diagnósticos académicos y quices, el diseño e implementación de solución analítica B.I para la Gestión de Indicadores Académicos, la implementación del flujo trabajo y trazabilidad para el sistema de información web para la gestión de contratos, el Sistema de Información para la Gestión de Proyectos Smart Campus, el diseño e implementación de

un sistema web para categorización docente (CATEDOC). Estas son algunas de las soluciones desarrolladas en el marco del proyecto de investigación.

De forma paralela, con el semillero SELECT se realizaron los siguientes proyectos: implementación de un prototipo para el control de acceso inteligente mediante tarjetas RFID y sensor biométrico para huella dactilar; el diseño de un dispositivo de medición inteligente para la monitorización del consumo energético; la implementación de un dispositivo de monitoreo para un sistema inteligente con sensores distribuidos para las variables de temperatura y humedad en el NOC de la UNIAJC y el desarrollo de un prototipo ecológico inteligente para generar una cultura de cuidado al medio ambiente en la comunidad universitaria. Con la realización de estos proyectos se han integrado estudiantes de programas de ingeniería electrónica y de tecnología en electrónica industrial, los cuales han sido guiados por profesores investigadores del grupo de investigación INTELIGO de la Facultad de Ingenierías.

6 Conclusiones y trabajos futuros

La priorización del proyecto en el marco del Plan Estratégico de Desarrollo de la universidad y la alineación de este con los objetivos estratégicos ha sido fundamental para el desarrollo de un ecosistema en la institución. Esto, por supuesto, ha requerido trabajar coordinadamente en el desarrollo de proyectos que articulan los ámbitos de desarrollo de un *Smart Campus*: Smart Living, Smart Environment, Smart People, Smart Economy, Smart Government y Smart Movility. Todos estos ámbitos han sido abordados desde el desarrollo de iniciativas que le apuntan al desarrollo de las funciones misionales y adjetivas de la universidad.

También, este proceso de transformación ha requerido la reestructuración de políticas al interior de la universidad. De la misma forma, la tecnología ha jugado un papel muy importante para lograr este ambicioso propósito en el que la participación de docentes y jóvenes investigadores ha jugado un papel muy importante. Ellos han sido los artífices de los resultados alcanzados hasta hoy, no sólo con la generación de productos de software, sino también con su participación en congresos mediante ponencias y publicaciones en revistas indexadas, que han permitido darle visibilidad al grupo GRINTIC, tanto en el ámbito nacional como internacional. Estos resultados obtenidos dan cuenta de la importancia de articular la investigación y la docencia como ejes transversales para el desarrollo del proyecto.

Asimismo, la adopción de un modelo de desarrollo basado en los marcos de desarrollo ágil como SCRUM y buenas prácticas de T.I. ha permitido consolidar un semillero de investigación con altas competencias humanas, que a la vez ha facilitado el fortalecimiento de capacidades institucionales para la producción de software, lo cual podría potenciar a futuro el desarrollo, innovación y emprendimiento, mediante la creación de Startup centradas en el desarrollo de productos y servicios empresariales que permitan avanzar de forma más rápida en el proceso de modernización y transformación digital de la universidad.

De la misma forma, el modelo de desarrolló permitió que los productos de software fueran creados con altos estándares de calidad, lo que a futuro facilitará el mantenimiento y la escalabilidad de las soluciones, para un crecimiento evolutivo efectivo y eficiente, dadas las necesidades cambiantes de la institución frente a nuevos requerimientos y mejoras que se puedan dar a futuro.

Los trabajos futuros para desarrollar en la tercera fase del proyecto de investigación estarán encaminados a fortalecer las capacidades institucionales para el desarrollo de proyectos orientados a mitigar la deserción académica de estudiantes, al fortalecimiento de las plataformas de interacción para la atención de servicios universitarios, la integración de servicios a través del desarrollo de una mesa de ayuda y la modernización de la plataforma de comunicaciones empresarial. Adicionalmente, se invertirán recursos para monitorear el aforo de ingreso a la universidad, que permita a futuro analizar variables tales como: ausencia intersemestral, control de asistencia de estudiantes, docentes y personal administrativo.