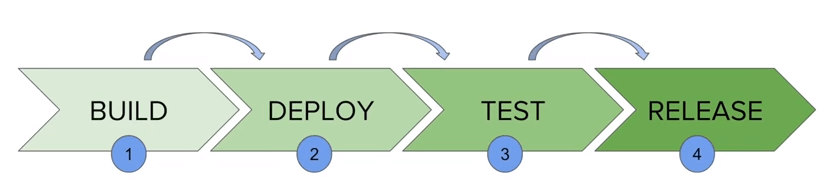
**Tutorial de Jenkins Pipeline: Cómo crear JenkinsFile (ejemplo)**

**¿Qué es Jenkins Pipeline?**

**Jenkins Pipeline** es una combinación de complementos que admite la integración e implementación de canalizaciones de entrega continua. Tiene un servidor de automatización extensible para crear canalizaciones de entrega simples y complejas como código a través de la canalización DSL. Un Pipeline es un grupo de eventos interconectados entre sí en una secuencia.

**¿Qué son las canalizaciones de entrega continua? ¿Cómo funciona?**

En una canalización de Jenkins, cada trabajo o evento tiene algún tipo de dependencia de al menos uno o más eventos.

[](https://www.guru99.com/images/1/091318_0510_JenkinsPipe1.png)Funcionamiento de las tuberías de entrega continua de Jenkins

La imagen de arriba representa una tubería de entrega continua en Jenkins. Contiene un grupo de estados denominados compilar, implementar, probar y liberar. Estos eventos están interrelacionados entre sí. Cada estado tiene sus eventos, que funcionan en una secuencia denominada canalización de entrega continua.

Una canalización de entrega continua es una expresión automatizada para mostrar su proceso de obtención de software para el control de versiones. Por lo tanto, cada cambio realizado en su software pasa por una serie de procesos complejos en su camino hacia su lanzamiento. También implica desarrollar el software de una manera confiable y repetible, y la progresión del software construido a través de múltiples etapas de [prueba](https://www.guru99.com/software-testing-introduction-importance.html) e implementación.

## ¿Qué es un JenkinsFile?

Las canalizaciones de Jenkins se pueden definir mediante un archivo de texto llamado **JenkinsFile.**Puede implementar la canalización como código usando JenkinsFile, y esto se puede definir usando un lenguaje específico de dominio (DSL). Con JenkinsFile, puede escribir los pasos necesarios para ejecutar una canalización de Jenkins.

Los beneficios de usar **JenkinsFile son** :

* Puede crear canalizaciones automáticamente para todas las ramas y ejecutar solicitudes de extracción con solo un **JenkinsFile.**
* Puede revisar su código de Jenkins en la canalización
* Puede auditar su canalización de Jenkins
* Esta es la fuente única de su canalización y varios usuarios pueden modificarla.

JenkinsFile se puede definir mediante la interfaz de usuario web o con un archivo Jenkins.

**Sintaxis de canalización declarativa frente a secuenciada:**

Hay dos tipos de sintaxis de canalización de Jenkins que se utilizan para definir su JenkinsFile.

1. Declarativo
2. Con guión

**Declarativo:**

La sintaxis declarativa de canalización ofrece una forma sencilla de crear canalizaciones. Contiene una jerarquía predefinida para crear canalizaciones de Jenkins. Le brinda la capacidad de controlar todos los aspectos de la ejecución de una canalización de una manera simple y directa.

**Con script:**

La canalización de Jenkins con guión se ejecuta en el maestro de Jenkins con la ayuda de un ejecutor ligero. Utiliza muy pocos recursos para traducir la canalización en comandos atómicos. Tanto la sintaxis declarativa como la escrita son diferentes entre sí y se definen de manera totalmente diferente.

## ¿Por qué utilizar la pipeline de Jenkin?

Jenkins es un servidor de integración continúa abierto que tiene la capacidad de soportar la automatización de los procesos de desarrollo de software. Puede crear varios trabajos de automatización con la ayuda de casos de uso y ejecutarlos como una canalización de Jenkins.

Estas son las razones por las que debe usar la canalización de Jenkins:

* La canalización de Jenkins se implementa como un código que permite a varios usuarios editar y ejecutar el proceso de la canalización.
* Las pipeline son robustas. Entonces, si su servidor se reinicia inesperadamente, la canalización se reanudará automáticamente.
* Puede pasar el proceso de canalización y esperar a que se reanude hasta que haya una entrada del usuario.
* Jenkins Pipelines apoya grandes proyectos. Puede ejecutar varios trabajos e incluso utilizar canalizaciones en un bucle.

## Conceptos de Jenkins Pipeline

|  |  |
| --- | --- |
| **Término** | **Descripción** |
| Pipeline | La canalización es un conjunto de instrucciones que se dan en forma de código para la entrega continua y consta de las instrucciones necesarias para todo el proceso de construcción. Con la canalización, puede crear, probar y entregar la aplicación. |
| Nodo | La máquina en la que se ejecuta Jenkins se llama nodo. Un bloque de nodo se utiliza principalmente en la sintaxis de canalización con script. |
| Stages | Un bloque de etapa contiene una serie de pasos en una canalización. Es decir, los procesos de compilación, prueba e implementación se unen en una etapa. Generalmente, se utiliza un bloque de etapa para visualizar el proceso de canalización de Jenkins. |
| Step | Un paso no es más que una sola tarea que ejecuta un proceso específico en un momento definido. Una tubería implica una serie de pasos. |

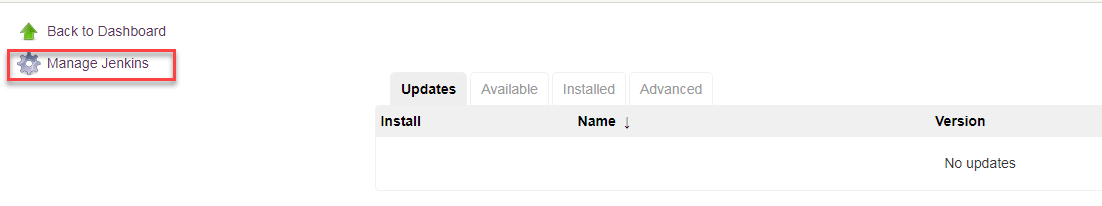
## Instalar el complemento Build Pipeline en Jenkins

Con el complemento de **canalización de compilación** , puede crear una vista de canalización de trabajos entrantes y salientes, y crear activadores que requieran intervención manual.

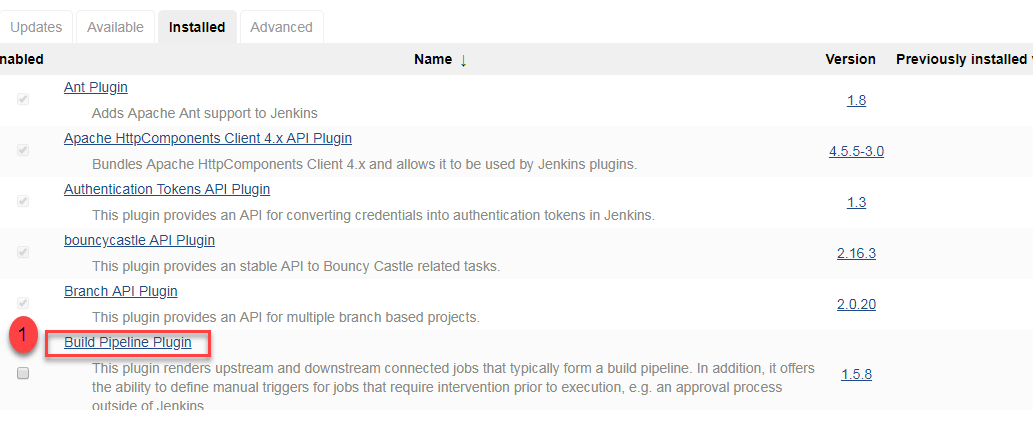
Así es como puede instalar el complemento de **canalización de compilación** en su Jenkins:

**Paso 1** ) La configuración del complemento se puede encontrar en,

**Administrar Jenkins> Administrar complementos.**

[](https://www.guru99.com/images/1/091318_0510_JenkinsPipe2.png)

Si ya ha instalado el complemento, se muestra en la pestaña de instalado.

[](https://www.guru99.com/images/1/091318_0510_JenkinsPipe3.png)

**Paso 2** ) Si no tiene el complemento instalado previamente,

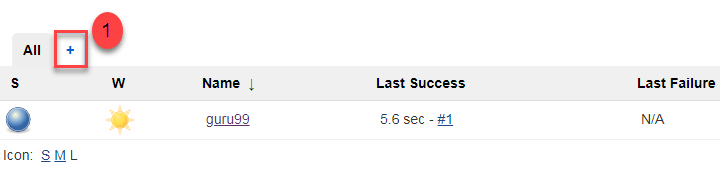
aparece en la pestaña **Disponible** .

Una vez que haya instalado correctamente el complemento de **canalización de compilación** en su Jenkins, siga estos pasos para crear su canalización de Jenkins:

## Cómo crear una canalización de Jenkins

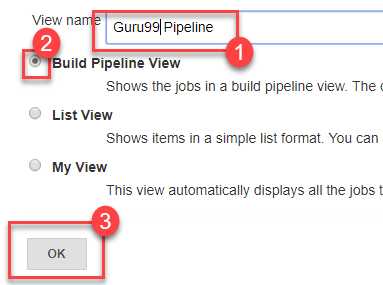
Una vez que haya iniciado sesión en su panel de Jenkins:

**Paso 1** ) Haga clic en el botón **"+"** en el lado izquierdo de su panel de Jenkins para crear una canalización.

[](https://www.guru99.com/images/1/091318_0510_JenkinsPipe4.png)

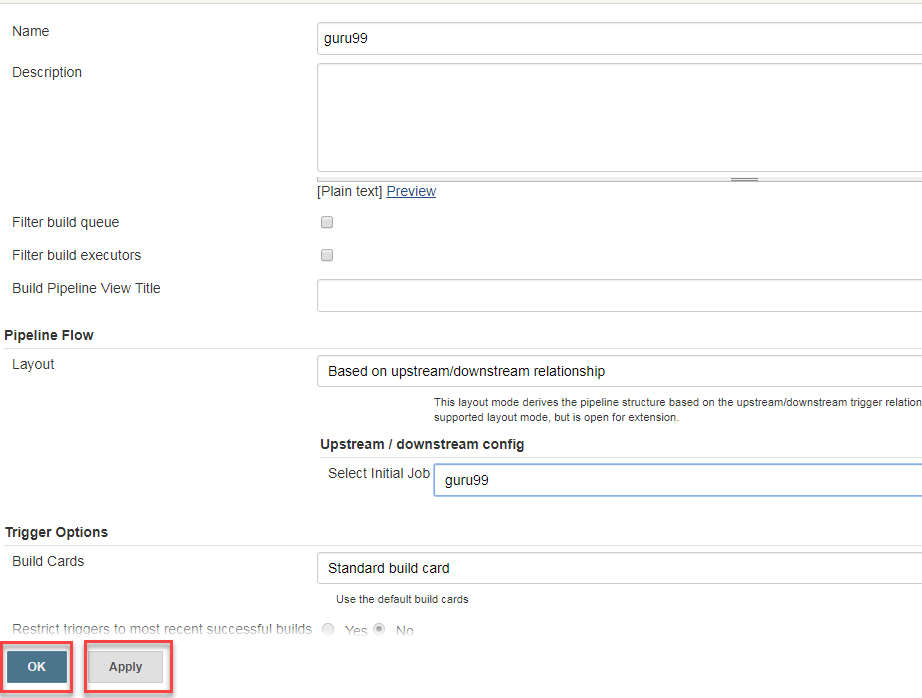
**Paso 2** )

1. Se le pedirá que dé un nombre a la vista de canalización. Lo llamaremos " **Guru99 Pipeline** " durante la duración de esta demostración.
2. Seleccione **Crear una vista de canalización** en **opciones.**
3. Haga clic en **Aceptar**

[](https://www.guru99.com/images/1/091318_0510_JenkinsPipe5.png)

**Paso 3** ) En la página siguiente, se le solicitarán más detalles para configurar su canalización de Jenkins. Simplemente acepte la configuración predeterminada y asegúrese de elegir el primer trabajo en la configuración.

Haga clic en **Aplicar** y luego en **Aceptar** .

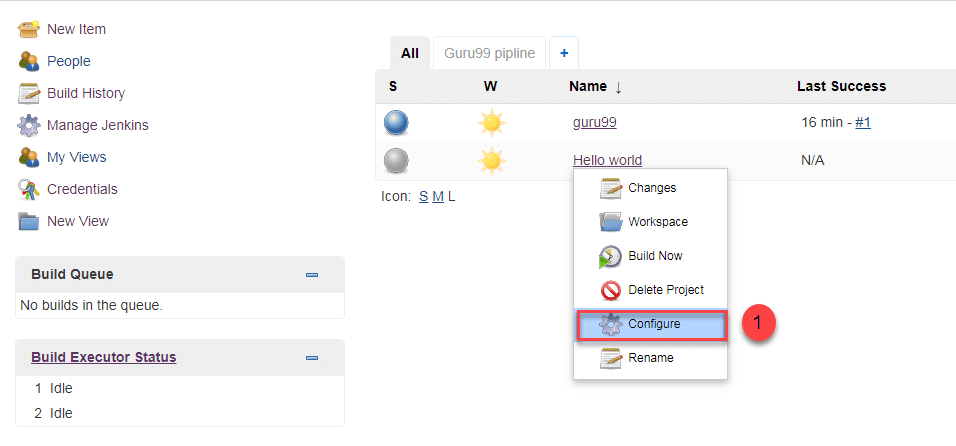
[](https://www.guru99.com/images/1/091318_0510_JenkinsPipe6.png)

Esto le mostrará la vista de canalización de muestra de su artículo, como se indica a continuación:

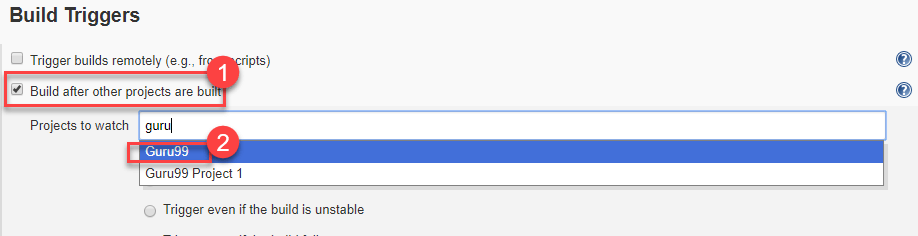
[](https://www.guru99.com/images/1/091318_0510_JenkinsPipe7.png)

## Ejecutando una compilación de Pipeline

**Paso 1** ) Para ejecutar una compilación de canalización, primero debe encadenar sus trabajos. Para ello, vaya a su primer trabajo y haga clic en configurar.

[](https://www.guru99.com/images/1/091318_0510_JenkinsPipe8.png)

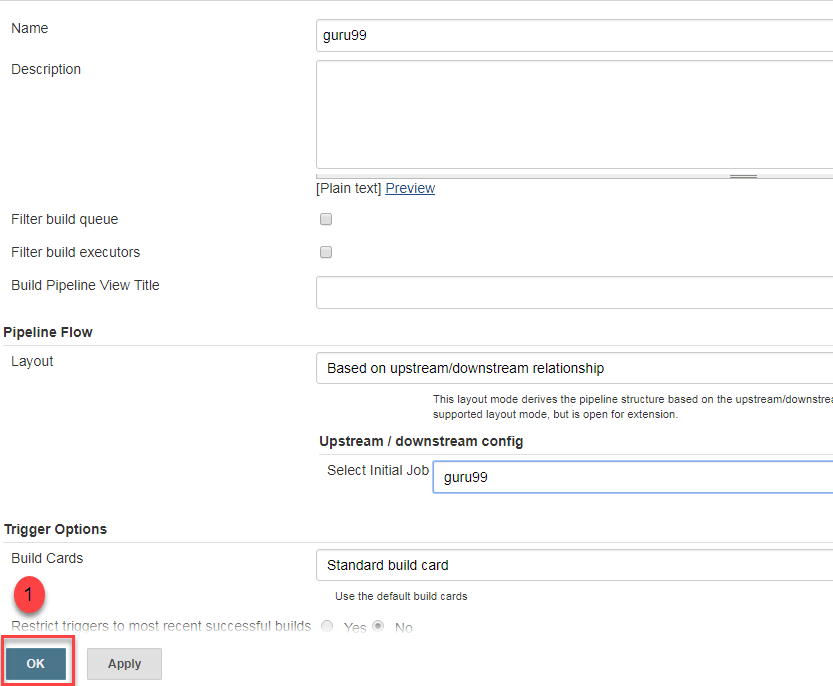
**Paso 2** ) Ahora, en **Build Triggers** , marque la opción **Build after other projects are built** .

[](https://www.guru99.com/images/1/091318_0510_JenkinsPipe9.png)

Así, se ha creado una cadena para todos sus trabajos.

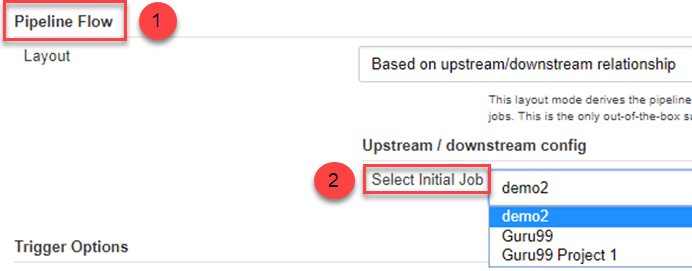
**Paso 3** ) Instale el complemento de **vista Build Pipeline** si aún no lo tiene instalado.

**Paso 4** ) Vaya a su panel de Jenkins y cree una vista haciendo clic en el botón " **+** ". Seleccione la opción **Crear vista de** canalización y haga **clic** en **Aceptar** .

[](https://www.guru99.com/images/1/091318_0510_JenkinsPipe10.png)

**Paso 5** ) En **Configuración de vista de canalización** , busque **Flujo de canalización** .

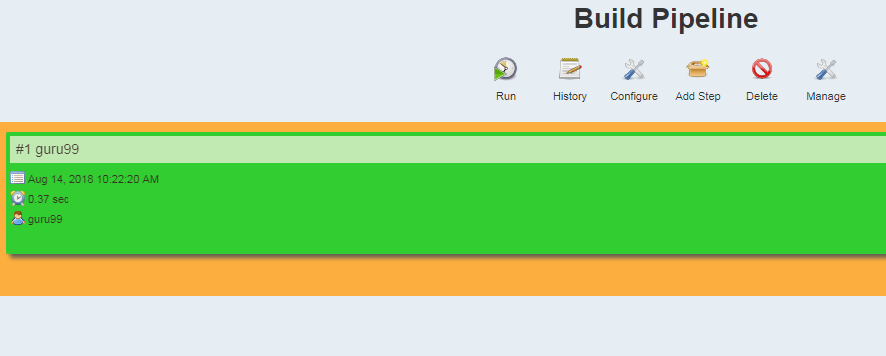
En **Flujo de canalización** , seleccione el trabajo inicial que se ejecutará. Ahora elija el trabajo que tiene cadenas a otros trabajos, como se muestra en el **Paso 1** y el **Paso 2** .

[](https://www.guru99.com/images/1/091318_0510_JenkinsPipe11.png)

Aquí hemos seleccionado **Guru99 Project 1** como el trabajo inicial, encadenado a otros trabajos. Entonces, uno por uno, los trabajos se ejecutarán en la canalización.

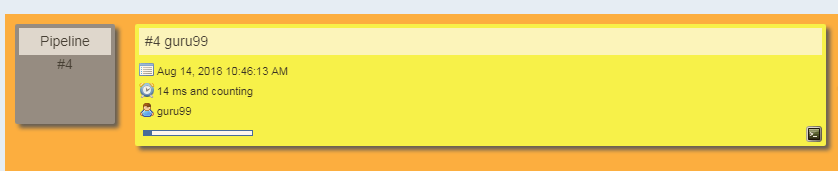
Cuando la tubería de Jenkins se está ejecutando, puede verificar su estado con la ayuda de los símbolos de estado rojo y verde. El rojo significa que la tubería ha fallado, mientras que el verde indica el éxito.

En este ejemplo de canalización de Jenkins, vemos que el botón es verde. Por lo tanto, la canalización es exitosa.

[](https://www.guru99.com/images/1/091318_0510_JenkinsPipe12.png)

## Ejecución de la pipeline de Jenkins

Haga clic en **Ejecutar** para ejecutar la canalización de Jenkins. Se verá algo como esto:

[](https://www.guru99.com/images/1/091318_0510_JenkinsPipe13.png)

En el ejemplo de secuencia de comandos de la canalización de Jenkins anterior, estamos demostrando un programa simple "helloworld.java". Pero en proyectos en tiempo real, será responsable de crear y construir tuberías complejas en Jenkins. Consulte a continuación para ver una vista de canalización de muestra.

[](https://www.guru99.com/images/1/091318_0510_JenkinsPipe14.png)