**Datos de identificación del programa de formación**

| PROGRAMA DE FORMACIÓN | Desarrollo de aplicaciones web full stack |
| --- | --- |

| COMPETENCIA | 220501097-Implementar la solución de software de acuerdo con los requisitos de operación y modelos de referencia. | RESULTADOS DE APRENDIZAJE | 220501097- 02 - Desplegar el *software* de acuerdo a los requerimientos técnicos. |
| --- | --- | --- | --- |

| NÚMERO DEL COMPONENTE FORMATIVO | 16 |
| --- | --- |
| NOMBRE DEL COMPONENTE FORMATIVO | Despliegue de *software* |
| BREVE DESCRIPCIÓN | Este componente formativo está diseñado para brindar los conceptos y conocimientos orientados a la implementación del *software*, donde conocerá todos los pasos y herramientas a tener en cuenta para realizar este proceso de manera correcta. |
| PALABRAS CLAVE | AWS, AZURE, ambientes de producción, roles de despliegue, administración de bases de datos. |

| ÁREA OCUPACIONAL | 6 - VENTAS Y SERVICIOS |
| --- | --- |
| IDIOMA | Español |

**TABLA DE CONTENIDOS**

**1. Herramientas de despliegue**

**2. Ambientes de producción o desarrollo**

**3. Actividades, roles y tipos de despliegue**

**INTRODUCCIÓN**

| Cuadro de texto |
| --- |
| Apreciado aprendiz, bienvenido a este componente formativo que está orientado al trabajo con las diferentes herramientas de despliegue para la implementación de sistemas de información. Es importante tener en cuenta que el proceso de implementación y despliegue de un *software* es una de las tareas que demanda un grado de coordinación y comunicación eficiente, porque de esto dependerá el correcto funcionamiento del aplicativo para el usuario final. Otro aspecto importante es realizar una adecuada configuración de los ambientes de producción de la herramienta, porque en muchas ocasiones el ambiente de desarrollo y pruebas es controlado, pero cuando se llega a la puesta en marcha del sistema en el ambiente real, suceden inconvenientes, los cuales se deben solucionar teniendo en cuenta el marco de trabajo y las necesidades funcionales del usuario o cliente. Por este motivo trabajaremos los diferentes roles y sus responsabilidades en el proceso de implementación de la herramienta, un trabajo conjunto entre los diferentes autores para realizar el tipo de despliegue correcto para que la herramienta tenga su mayor eficiencia.  En el siguiente video conocerá, de forma general, la temática que se estudiará a lo largo del componente formativo. |

**GUION DE VIDEO INTRODUCTORIO**

| **Tipo de recurso** | Video motion | | | |
| --- | --- | --- | --- | --- |
| **NOTA** |  | | | |
| **Título** | Introducción | | | |
| **Escena** | **Imagen** | **Sonido** | **Narración** | **Texto** |
| **1** | Implantación de sistemas de la información 2022 |  | Bienvenido a este componente formativo donde podrá actualizar sus conocimientos sobre implementación del sistema de información, uno de los procesos que se deben llevar a cabo cuando se desarrolla un proyecto de *software*. Esto implica un grado de conocimientos técnicos y de acoplamiento muy rigurosos, porque este proceso debe garantizar un adecuado funcionamiento del aplicativo construido y que, una vez implementado, funcione de manera efectiva para el cliente que solicitó su construcción. Para este propósito se deben utilizar herramientas, procesos y procedimientos que garanticen un correcto despliegue del mismo. Actualmente existen herramientas y dispositivos para el despliegue del proceso, lo que se llama computación en la nube y forma parte de las infraestructuras que se estudiarán. | Sistema de información  *Software*  Aplicativo  Implementar  Cliente  Herramientas  Nube |
| **2** |  |  | Los servicios en la nube son esenciales en los procesos de despliegue de aplicaciones; actualmente uno de estos servicios es el de AWS, que ofrece Amazon, donde se destaca el uso de máquinas virtuales y el acceso a bases de datos de alta concurrencia. Una ventaja de estos servicios es el bajo costo que se requiere para utilizarlos; anteriormente, las empresas utilizaban servidores propios e infraestructura costosa para el despliegue de sus servicios y aplicativos y ahora solo basta contratar los servicios de *Amazon Web Services* (AWS) y se obtiene toda una infraestructura a nuestra disposición para implementar cualquier tipo de solución informática actualizada y confiable para muchas empresas. | Aplicaciones  AWS  Máquinas virtuales  Servidores  Infraestructura  Servicios |
| **3** |  |  | Microsoft Azure es una plataforma que incluye lo mejor en servicios de despliegue y la puesta en marcha de servicios y de *software.* Esta herramienta tiene una alta integración con diferentes plataformas y lenguajes de programación que la convierte en una de las mejores alternativas para realizar los procesos de despliegue de sistemas de información;.Otro de los aspectos que la convierten en una ventaja competitiva es el respaldo que posee de Microsoft, debido a que constantemente actualiza los servicios y componentes que se ofrecen a través de esta plataforma. Es importante resaltar que también cuenta con diferentes certificaciones, entre ellas la de ISO 27001 que refleja todos los aspectos de seguridad de la información. | Microsoft Azure  *Software*  Herramienta  Microsoft  ISO 27001 |
| **4** |  |  | Los roles y responsabilidades en un proyecto de *software* generan dinamismo e inercia a cualquier equipo de trabajo, pero es fundamental, de acuerdo con la experiencia y habilidades de cada miembro del equipo de trabajo, asignar actividades, tareas y responsabilidades en el marco de esas condiciones. Uno de los problemas que se puede presentar al interior de un equipo de trabajo es no tener una definición clara de los roles y responsabilidades de sus miembros; lo que puede generar sobrecargas de trabajo es ubicar personas con procesos y procedimientos para los cuales no poseen la experiencia y conocimientos para hacerlo. Esto abre las puertas a los errores y riesgos a los cuales se ve expuesto el proyecto y por eso la importancia de una asignación de roles de manera adecuada para evitar problemas más adelante. | Equipo de trabajo  Habilidades  Roles  Proyecto |
| **5** |  |  | Conscientes de que su disciplina y estudio de esta temática generará mejores resultados en el cumplimiento de sus metas, le deseamos éxito en su trabajo profesional. | Disciplina  Temática |
| **Nombre del archivo** | 228125\_v1 | | |  |

**DESARROLLO DE CONTENIDO**

**1 Herramientas de despliegue**

| Cuadro de texto |
| --- |
| DevOps ingenieros diseño web con gente. Los desarrolladores de DevOps trabajan con la escena de los ordenadores. Composición de las operaciones de desarrollo en forma plana. Ilustración vectorial para materiales promocionales de medios sociales.  Comencemos hablando de las herramientas de despliegue, las cuales permiten realizar uno de los procesos que tienen mayor importancia en el desarrollo de aplicaciones informáticas, realizar un seguimiento y mantenimiento del código fuente del aplicativo. Adicional a ello, permiten la integración de módulos del sistema como también la puesta en marcha de los aplicativos de *software* desarrollados bajo la premisa de cobros, pero siempre conservando un mejor nivel adquisitivo para el usuario o cliente final del aplicativo. |

| Cuadro de texto |
| --- |
| Antes de continuar, respondamos la siguiente pregunta: ¿qué es el proceso de integración continua?Icono de pregunta signo/símbolo de consulta de ilustración vectorial plana. Para la interfaz de usuario móvil  Es una práctica habitual en desarrollo de *software* que consiste en integrar frecuentemente mejoras en el código de un proyecto una vez han sido validadas, normalmente varias veces al día, con el objetivo de detectar errores lo antes posible.  Cada cambio que realiza un desarrollador (sea una resolución de un *bug*, la creación de una nueva funcionalidad, etc.), se comprueba compilando el código fuente y obteniendo un ejecutable (llamado *build*). Si es validado, será incorporado al código fuente y desplegado. |

| **Tipo de recurso** | Infografía interactiva Modal | |
| --- | --- | --- |
| **Texto introductorio** | Teniendo claro el concepto anterior, conozcamos algunas herramientas de despliegue: | |
| Los textos y enlaces se encuentran en la carpeta Anexos con el nombre Infografia1.pptx | | |
| **Código de la imagen** | 228125\_i3 | |
| **Punto modal 1** | Es una herramienta para realizar los procesos de integración continua en proyectos de desarrollo de *software*, adicional a ello, permite realizar pruebas para garantizar el correcto funcionamiento del aplicativo. Funciona como un servidor centralizado que provee los servicios de alojamiento de código fuente y permite que varios usuarios puedan trabajar en él de manera simultánea, garantizando que pueda ser utilizado por equipos de desarrollo en todo el mundo, Con Jenkins, las organizaciones aceleran el proceso de desarrollo y entrega de *software* a través de la automatización. Mediante sus centenares de *plugins*, se puede implementar en diferentes etapas del ciclo de vida del desarrollo, como la compilación, la documentación, el testeo o el despliegue. | Jenkins |
| **Punto modal 2** | Es un sistema de Integración continua, gratuita para proyectos *Open Source* y de pago para proyectos privados. Se integra sin problemas con GitHub y automáticamente ejecuta el *pipeline* definido en cada *push* o *pull requests,* el cual permite el testeo de aplicaciones codificadas en Ruby, Node, Objective-C, Go, Java, C# y F# entre otros lenguajes de programación. Una de las particularidades que posee es que puede ser conectado o integrado a los repositorios de código que se encuentren alojados en *github*, dándole un control abierto al equipo de desarrollo de gestionar las operaciones realizadas en el código elaborado, aunque esto no se pueda ver interesante desde un principio si lo vemos a fondo, genera de alguna manera, competitividad a los equipos de desarrollo.  Cuenta con aproximadamente 300.000 proyectos y 235.000 usuarios probando los mismos. Estos números hablan sobre la importancia de la herramienta y lo robusta que es. Agregado a esto, algunas de las compañías más importantes que utilizan Travis son Facebook, Heroku, Mozilla, Rails, Twitter, Zendesk, Slack.  En la página oficial se puede encontrar documentación paso a paso con variados lenguajes. Además, Travis-CI cuenta con una comunidad activa muy grande, por ende encontrar soluciones a problemáticas es bastante común. | Travis CI |
| **Punto modal 3** | Es una herramienta que se utiliza para el despliegue de sistemas de información de diferentes tipos, está soportada por la empresa Amazon, la cual ofrece los servicios de bases de datos relacionales y no relacionales, subir servicios para páginas web y dar soporte a las mismas, creación de máquinas virtuales, manejo de inteligencia artificial, entre otros servicios. Esta plataforma ha tomado bastante fuerza en los últimos años debido a sus precios competitivos que se ajustan de acuerdo con la necesidad del cliente o empresa que requiere dichos servicios. Otro de los aspectos que se debe tener en cuenta es que está empresa cuenta con diferentes certificaciones en el ámbito de seguridad de la información y soporte técnico, lo cual es una ventaja, porque si lo vemos desde el punto de vista de infraestructura, las empresas anteriormente tenían que mantener los costos de una infraestructura compleja 24/7 y contratar personal altamente calificado para mantenerla; en términos de costos esto es bastante enorme, sin decir de las certificaciones de operación. El servicio de Amazon entra dentro de la tendencia *cloud computing,* la cual tiene como objetivo ofrecer esa misma infraestructura pero por fuera de la empresa y para que esta pueda acceder a la misma, lo que requiere es una buena conectividad en términos de mantenimiento es una ventaja competitiva y ni decir del soporte, porque al momento en que se contrata este tipo de servicios, se incluye el soporte técnico, el cual es de primera calidad. | AWS |
| **Punto modal 4** | Es un producto de Microsoft que proporciona funciones de control de versiones, informes, gestión de requisitos, gestión de proyectos, compilaciones automatizadas, pruebas y gestión de versiones. Cubre todo el ciclo de vida de la aplicación y habilita las capacidades de DevOps; una de las características fundamentales de este servicio es el control de versionamiento, el cual permite realizar un seguimiento bastante detallado de cada movimiento que se realiza dentro del proyecto. Regularmente, cuando se realiza seguimiento al proyecto, se utilizan diferentes herramientas para lograr un control de manera completa, en este caso, esta herramienta integra lo mejor de todas esas herramientas en un solo sitio, lo cual permite que se puedan realizar diferentes tareas y facilitar el proceso de control del proyecto en cualquier momento.  Los partidarios de DevOps describen varias ventajas técnicas y empresariales con las que, en última instancia, se consiguen clientes más satisfechos. Entre algunas de las ventajas de DevOps se incluyen las siguientes (NetApp, s.f.):   * Una mejor y más rápida entrega de productos. * Resolución de problemas en menos tiempo y con menor complejidad. * Mejor escalabilidad y disponibilidad. * Entornos de funcionamiento más estables. * Mejor utilización de los recursos. * Mayor automatización. * Mayor visibilidad de resultados del sistema. * Mayor innovación. | Azure DevOps |
| **Punto modal 5** | Es una plataforma que se aloja en la web y tiene como propósito brindar servicios *GIT* y *DEVOPS*, para que desarrolladores e integradores de código puedan realizar pruebas y despliegue de código fuente a nivel general. Esta plataforma también tiene la posibilidad de realizar control de código, lo cual es una ventaja bastante significativa, debido a que integra lo mejor de las herramientas de integración, lo cual permite que en una misma *suite* se tenga la posibilidad de realizar todas estas operaciones en un mismo lugar.  Gitlab es un *software* que va mejorando y creciendo con diversas funcionalidades interesantes que son listadas a continuación (Turingears, s.f.):   * Permite autentificar servicios como LDAP. * Tiene distintos tipos de acceso y permisos (uso de roles y grupos) y restringiendo a ciertos usuarios. * Los usuarios pueden acceder al proyecto a través de la web y por SSH. * Seguimiento de incidencias y comentarios de un proyecto. * Capacidad para importar repositorios existentes, ya sea GitHub u otros. * Interfaz web para interaccionar con GitLab permitiéndonos trabajar de una manera intuitiva. * Historial de modificaciones del proyecto. | GitLab |

| Cuadro de texto |
| --- |
| Continuemos hablando un poco más sobre GitLab. La ventaja principal es que facilita notablemente el desarrollo de *software* ágil entre varios equipos. De esta manera, varios desarrolladores pueden trabajar simultáneamente en un proyecto y editar, por ejemplo, diferentes funciones de forma paralela. La protocolización continua de todos los procesos, garantiza que no se pierda ninguna modificación del código ni que se sobreescriba de forma no intencionada. También es posible deshacer rápidamente los cambios ya aplicados, en esta nueva tendencia del agilismo, donde las metodologías de desarrollo ágil como Scum, Kanban y XP permiten la integración del usuario en los procesos de desarrollo del producto de *software*. Así se garantiza que el proceso de integración se realice de manera mucho más detallada y con el cliente, teniendo en cuenta que es el protagonista en el proceso de construcción del producto.  Entre las funciones más importantes de GitLab se encuentran las siguientes:    Ryazan, Russia - June 26, 2018: Homepage of GitLab website on the display of PC. URL - GitLab.com.   * Interfaz fácil de usar. * Las ramificaciones pueden permanecer privadas o publicarse. * Posibilidad de gestionar varios repositorios. * Revisión de códigos. * Localización integrada de errores y problemas. * Integración continua y entrega continua integradas de forma gratuita. * Wikis de proyectos. * Creación sencilla de fragmentos de código para dividir partes del código. |

| **Tipo de recurso** | Tarjetas Avatar | |
| --- | --- | --- |
| **Introducción** | GitLab está basado en un código fuente abierto y de acceso libre. En el año 2013 se creó una versión *Enterprise* propia para empresas, por lo que ahora hay dos modelos de uso: | |
| GitLab CE: Community-Edition (gratuita). | | r/gitlab - Looking for Gitlab 11.11 links...  **Imagen** 228125\_i5 |
| GitLab EE: Enterprise-Edition (de pago). | | GitLab Enterprise Edition  **Imagen:** 228125\_i6 |

| Tipo de recurso | Cajón de texto de color |
| --- | --- |
| Ambas versiones están basadas en la licencia de código abierto MIT. La Enterprise-Edition cuenta con algunas funciones adicionales en comparación con la Community. En este contexto, GitLab ofrece tres modelos de suscripción en función del alcance de las funciones adicionales deseadas. | |

**2 Ambientes de producción o desarrollo**

| Cuadro de texto |
| --- |
| Ahora, en el desarrollo de los productos de *software* intervienen diversas etapas, una de ellas es la codificación de los requerimientos funcionales y no funcionales, conocida como desarrollo; posteriormente es la revisión del cumplimiento de dichos requerimientos en el *software* y la usabilidad que muestra para los usuarios, que se conoce como aseguramiento de la calidad o pruebas. Por último, es puesto en operación para brindar un servicio a los usuarios que lo utilizarán.Programación para desarrolladores de software, búsqueda de soluciones mientras se trabaja en equipos de escritorio en la sala de control de sistemas del centro de datos. Equipo de jóvenes profesionales haciendo codificación de alta tecnología  El ambiente de desarrollo se usa para integrar el código de los diferentes componentes del equipo de programación, así como los elementos que forman el sistema de información o aplicación, como son: lenguajes, librerías, *frameworks*, base de datos, entre otros. De igual manera, se puede hacer uso de herramientas que apoyen al entorno de desarrollo, como por ejemplo, el Subversion o el GitLab, que ayudan a gestionar el código fuente, para el control de versiones y su reutilización. |

| **Tipo de recurso** | Infografía interactiva Punto caliente | |
| --- | --- | --- |
| **Texto introductorio** | Conozcamos los diferentes tipos de entorno: | |
|  | | |
| **Código de la imagen** | 228125\_i8 | |
| **Punto caliente 1** | El entorno de desarrollo propiamente dicho se refiere a la creación del *software*. Puede ser manejado a nivel local en el ordenador del programador o en un servidor local. Así mismo, puede ser procesado en herramientas desarrolladas para este tipo de tareas.  Es habitual que trabaje un equipo en el desarrollo de un proyecto, lo que hace que se realicen tareas de forma individual. El entorno de integración facilita la unión de los diferentes desarrollos y permite comprobar que los diferentes trabajos no interfieran entre sí. | **Entorno de desarrollo** |
| **Punto caliente 2** | Este tipo de entorno usualmente está ubicado en la nube. La ventaja de contar con un servidor de pruebas, es que permite a otros miembros del equipo y clientes a interactuar con el *software* para realizar *tests* de funcionalidades del sistema.  También se pueden realizar *testing* automatizados, su principal objetivo es detectar el mayor número de fallos posible, al no necesitar una persona que realice un testeo manual, permite ahorrar tiempo. Además, la ejecución de pruebas automatizadas permite ejecutar *tests* durante las 24 horas del día los 7 días de la semana. | **Entorno de pruebas o *testing*** |
| **Punto caliente 3** | También llamado entorno de *staging*, permite trabajar en un entorno con una configuración exactamente igual a la que existe en el entorno de producción. La finalidad de usar este tipo de servidor es simular el entorno de producción, para validar su usabilidad en un entorno real. Del mismo modo, permite poder realizar pruebas de actualizaciones y minimizar las causas que puedan generar caídas del sistema. | **Entorno de pre-producción** |
| **Punto caliente 4** | Es el entorno donde finalmente se ejecuta el *software*, el cual es utilizado por el usuario Este servidor, a diferencia del servidor de pre-producción, debería tener una mayor infraestructura y mayor capacidad de manejo de tráfico o de conexiones recurrentes.  Si el entorno de producción está bien configurado y se han realizado pruebas automatizadas y por parte del usuario, no debería existir ninguna incidencia en la ejecución del *software* final. | **Entorno de producción** |

| Cuadro de texto |
| --- |
| Podemos así concluir que los entornos de producción son ambientes que poseen una tecnología que permite realizar el despliegue de un aplicativo de *software* de una manera masiva para ser utilizado por varios usuarios al tiempo, es donde se incluyen los datos de operación y empieza a ser alimentado con datos reales y en condiciones reales de operación. Esto es algo que se debe realizar mediante la transición del entorno de desarrollo hasta llegar al entorno de producción donde el *software* será usado masivamente. |

| **Tipo de recurso** | Slider Imagen | |
| --- | --- | --- |
| **Introducción** | A continuación, se presentan unos esquemas de esta transición. | |
| **Entorno de desarrollo**  El entorno de desarrollo utiliza áreas de proyecto en el almacén de objetos de diseño para aislar los efectos del restablecimiento del entorno de pruebas. Cada área de proyecto tiene su propio almacén de objetos de destino y su propia región de *Content Platform Engine.* | | Nota. Tomado de IBM Documentación (2022).  **Imagen:** 228125\_i9 |
| **Entorno de prueba**  Es para probar las soluciones que se han creado en el entorno de desarrollo. Puede tener varios entornos de pruebas, por ejemplo, un entorno de pruebas de rendimiento y un entorno de pruebas de preproducción. | | Nota. Tomado de IBM Documentación (2022).  **Imagen:** 228125\_i10 |
| **Entorno de producción**  El entorno de producción está en un dominio de FileNet P8 diferente. En el entorno de producción necesita un único almacén de objetos de transferencia a donde se importan y desde donde se despliegan las soluciones. También necesita entornos de destino donde desplegar las soluciones. Puede desplegar muchas soluciones en un único entorno de destino. Asegúrese de tener suficientes recursos de sistema para el número de casos que prevé tener en cada base de datos. | | Nota. Tomado de IBM Documentación (2022).  **Imagen:** 228125\_i11 |

| Cuadro de texto |
| --- |
| Pudimos observar los diferentes procesos que debe sufrir un aplicativo desarrollado antes de llevar a producción, donde el usuario final tendrá la oportunidad de probar el sistema y se empieza su prueba real; este proceso de transición no es fácil, por ello se deben seguir todos los pasos y procedimientos necesarios para llevar a cabo de manera eficiente todo lo que se requiera.  Para conocer cuáles son las técnicas utilizadas para los procesos de producción, le invitamos a ver el siguiente video. |

| **Tipo de recurso** | Video motion | | | |
| --- | --- | --- | --- | --- |
| **NOTA** |  | | | |
| **Título** | Técnicas utilizadas para los procesos de producción | | | |
| **Escena** | **Imagen** | **Sonido** | **Narración** | **Texto** |
| **1** |  |  | Las técnicas utilizadas para los procesos de producción se refieren a la separación de las capas o componentes del *software* a desarrollar, es decir, la reparación entre el *front-end* y el *back-end*. Esta es una excelente manera de trabajar proyectos de *software* porque se garantiza la calidad del código, la separación funcional de cada componente y se realiza la depuración e integración del código de una manera mucho más eficiente. Dentro de los procesos realizados en los entornos de producción se debe garantizar que una vez se despliegue el aplicativo en un entorno real de producción, con usuarios y datos reales, si se desea realizar algún cambio en las funcionalidades y proceso del *software*, este no genere tanto traumatismo y detenga los procesos que se encuentran en ejecución. | Procesos de producción  *Software*  Calidad del código  Componente  Producción  Datos |
| **2** |  |  | Existe una gran diferencia entre entorno de desarrollo y entorno de producción y es, precisamente esta diferencia, la que marca la pauta del proceso de puesta en marcha de aplicaciones a nivel mundial, Cuando se habla de entornos de desarrollo, son aquellos que tienen una ejecución local en la máquina del desarrollador, es decir, todos los componentes como base de datos, código fuente y ejecución del mismo, se llevan a cabo en la máquina de quien trabaja el proceso de codificación y los datos, que se utilizan de prueba, son controlados por el desarrollador, lo que se denomina pruebas iniciales. Sin embargo, estas no contienen el contexto o formato específico de prueba, sino que son para verificar que el programa funcione inicialmente, a diferencia de un entorno de producción donde el aplicativo no se encuentra en la máquina local del desarrollador, sino que se encuentra en un repositorio, e instalado en un ambiente de producción, lo que hace que los usuarios reales del aplicativo realicen el uso del *software* utilizando datos reales. Esto, de alguna manera, cambia las reglas del juego porque dependemos de un tercero que presta el servicio de intermediación entre los usuarios del sistema y el aplicativo como tal. | Desarrollo  Producción  Ejecución  Base de datos  Código fuente  Codificación  Pruebas iniciales |
| **3** |  |  | Cuando se hace referencia a solicitudes de usuarios estas son peticiones que se reflejan en la solicitud de información que cumplen ciertas funciones reflejadas en el CRUD de una base de datos, es decir, está conectada directamente con la creación de nuevos registros, búsqueda, actualización y eliminación de los mismos. | Usuarios  Información  CRUD |
| **4** |  |  | Para poder acceder a los objetos de una base de datos se necesita tener privilegios (permisos). Los privilegios sobre objetos permiten acceder y realizar cambios en las tablas de los esquemas de la base de datos. Por ejemplo, *INSERT, SELECT, UPDATE* y *DELETE*, son privilegios sobre objetos. | Objetos  Permisos  Tablas |
| **5** | Ambientes de trabajo para software: La forma básica para desarrollar,  probar y lanzar - ortizvivas.com |  | Los permisos de uso de las instrucciones SQL para crear nuevos objetos de la base de datos son los permisos para ejecutar algunas sentencias SQL. Un usuario que dispone de estos permisos puede, por ejemplo, crear sus propias tablas y procedimientos. La asignación de estos permisos no debería poder llegar a ser peligrosa y, como para todos los permisos, se deben asignar solo cuando sea necesario. | SQL  Objetos  Base de datos  Usuario |
| **6** | Ambientes de pruebas integrales de software: Buenas prácticas |  | Permisos principales de instrucciones disponibles: *CREATE DATABASE – CREATE PROCEDURE – CREATE FUNCTION – CREATE TABLE – BACKUP DATABASE.* | Permisos  Instrucciones |
| **Nombre del archivo** | 228125\_v2 | | |  |

| Cuadro de texto |
| --- |
| Lo mencionado anteriormente está muy conectado con permisos de manipulación de la estructura de la base de datos, lo cual solo debe permitirse al administrador de la base de datos y por ningún motivo debe el usuario final modificar, a menos que se requiera bajo algún procedimiento a realizar. |

| Cuadro de texto |
| --- |
| Hablemos ahora sobre los reportes, los cuales son estructuras que se encuentran estrechamente relacionados con la información almacenada en la base de datos del aplicativo; en otras palabras, son consultas que se realizan para obtener datos almacenados en un sistema de información con una configuración y estructura específica, esto se realiza con el propósito de obtener información ordenada y que pueda servir para la toma de decisión en una organización. Los reportes a menudo mezclan diferentes aspectos de la información de una organización, es decir, un reporte puede incluir información general, financiera, presupuestal y de recursos humanos al tiempo para tener una comprensión y datos organizados con un sentido propio. |

| **Tipo de recurso** | Tarjetas Avatar | |
| --- | --- | --- |
| **Introducción** | Un ejemplo de ello es cuando una organización desea conocer las facturas que se han emitido en un periodo específico, cada una de estas facturas trae: | |
| Información del cliente a quien se le realizó dicha factura. | | icono de persona  **Imagen** 228125\_i12 |
| Información de los productos que se encuentran incluidos en dichas facturas. | | productos icono gratis  **Imagen:** 228125\_i13 |
| Información de precios, cálculo de subtotales, impuestos y el total de la factura. | | Icono de transferencia de dinero redondo negro, flecha simple financiera marca en dólares venta diseño plano pictograma vector, elementos de interfaz infográfica para el logo de aplicación botón web ui ux aislado en fondo blanco  **Imagen:** 228125\_i14 |

| Cuadro de texto |
| --- |
| A esto es a lo que se le llama reporte: la agrupación de diferente tipo de información ubicada en un mismo espacio para interpretarla de una manera más sencilla.  Además, en ciertas ocasiones no se limitan a informar sobre una determinada cuestión, sino que también establecen objetivos, elementos persuasivos (como sugerencias o consejos) o incluso la conclusión de una determinada investigación. |

| **Tipo de recurso** | Tarjetas Avatar | |
| --- | --- | --- |
| **Introducción** | Es importante tener en cuenta que está información que se despliega en los mismos puede ser de tipo: | |
| **Digital**  Puede provenir de una base de datos, donde por medio de consultas, se obtiene esta información para mostrarla al usuario. | | Documento manual, tecnología de procesamiento de grandes datos, almacenamiento y análisis, rueda cognitiva y archivo de papel, captura de información digital, solución de software, icono de línea vectorial, ilustración lineal  **Imagen** 228125\_i15 |
| **Impresa**  Existe también la posibilidad de imprimir para tener un soporte físico de estos datos. | | Icono de documento de papel. Símbolo negro aislado. Ilustración vectorial sobre fondo blanco.  **Imagen:** 228125\_i16 |

| Cuadro de texto |
| --- |
| Las consultas son sentencias que se ejecutan en un servidor de bases de datos y que generan la información requerida por parte del usuario que la solicite; están conectadas con el lenguaje SQL que significa lenguaje de consulta estructurado, el cual utiliza sentencias y comandos para recuperar la información almacenada, teniendo en cuenta restricciones de datos y condiciones que se deben cumplir para entregarle al usuario final la información solicitada. En el caso de estar trabajando en un entorno de desarrollo la información vendrá del servidor de bases de datos que se esté ejecutando en la base de datos alojada de manera local en la máquina del programador, el cual tendrá acceso a los datos y resultados de manera única, pero en caso de tener desplegado el servidor alojado en una infraestructura de despliegue en la web, está información podrá ser consultada por millones de usuarios al tiempo.Tabla de bases de datos con almacenamiento de servidores y red en segundo plano del centro de datos |

| Cuadro de texto |
| --- |
| Por ejemplo, supongamos que tiene una tabla denominada Productos y otra tabla denominada Servicios. Ambas tablas tienen campos que contienen el nombre del producto o servicio, el precio, la disponibilidad de garantía del servicio o de garantía del producto, y si el producto o servicio se ofrece de forma exclusiva. Aunque en la tabla Productos se almacena información de garantía del producto y en la tabla Servicios se almacena información de garantía del servicio, la información básica es la misma (si un producto o servicio determinado incluye una garantía de calidad). Puede usar una consulta de unión, como la que se muestra a continuación, para combinar los cuatro campos de las dos tablas:  *SELECT name, price, warranty\_available, exclusive\_offer*  *FROM Products*  *UNION ALL*  *SELECT name, price, guarantee\_available, exclusive\_offer*  *FROM Services*  Como podemos observar, las consultas nos permiten recuperar la información que se encuentra dentro de la base de datos y luego poder realizar los procedimientos requeridos por la organización. |

**3 Actividades, roles y tipos de despliegue**

| Cuadro de texto |
| --- |
| Las actividades que se realizan en el proceso de despliegue de un sistema de información requieren que cada miembro del equipo de trabajo ejecute los procesos de acuerdo a su rol dentro del proyecto y adicional a ello cumpla con el cronograma de despliegue el cual establece las actividades, responsable y tiempos que se deben cumplir al realizar dicho proceso. |

| **Tipo de recurso** | Pestañas o tabs horizontales | |
| --- | --- | --- |
| **Introducción** | A continuación, veremos algunos procesos que se llevan a cabo para cumplir con este propósito. | |
| **Lanzamiento** | El lanzamiento es un proceso que se lleva a cabo para colocar en funcionamiento el aplicativo en el sistema o infraestructura de despliegue; este proceso inicia desde que se colocan los archivos que se encuentran de manera local dentro de la máquina del programador, hasta las pruebas de iniciación del aplicativo dentro de la nueva infraestructura, todo este proceso lleva consigo diferentes pasos que se deben realizar, por eso es aconsejable tener cada componente a lanzar de manera organizada. Un ejemplo de ello es: se desea lanzar una aplicación para inventarios que ya se encuentra desarrollada, por una parte, tenemos los archivos de código fuente, por otro lado, tenemos el archivo de *script* de la base de datos con los datos iniciales para la puesta en marcha del sistema; y por otro lado tenemos archivos de configuración y complementos que se deben tener en cuenta para realizar este proceso. Ahora bien, al tener todo esto no debemos dejar de lado la configuración del entorno de producción, el cual tiene como objetivo hacer accesible la aplicación para los usuarios que la requieran en todos estos aspectos; se hace una prueba inicial para verificar que todas las configuraciones y carga del sistema se realiza de manera correcta. | Plantilla de diseño de degradado de sitios web vectores. Desarrollo de software y codificación de aplicaciones. El ingeniero de software escribe código de computadora. Lanzamiento de conceptos de página para desarrollo de sitios web y móviles. Ilustración  Imagen: 228125\_i17 |
| **Instalación y activación** | La instalación y activación del sistema dependen en gran medida al tipo de sistema de información que se esté implementando, por ejemplo, en caso de que el aplicativo necesite copias, los archivos requeridos para su ejecución, en otras palabras, la instalación provee y coloca los archivos necesarios en el servidor de despliegue. En caso de que estemos utilizando una infraestructura como AWS o AZURE para desplegar nuestro aplicativo o en caso de estar realizando el proceso en una máquina, tal como un PC o dispositivo móvil, lo que hace este proceso es colocar los archivos que permiten que el *software* se ejecute de manera correcta; ahora, el proceso de activación depende de la negociación establecida para dar acceso al sistema, por ejemplo, licencias o permisos de uso, en este caso como bien sabemos las empresas y *software* en general, tienen ciertos procesos de acceder. Si vemos por ejemplo el caso del *software* de Office, una vez que se instala, este solicita un código de licencia para activar el sistema y ser utilizado de acuerdo con las políticas y tipo de licencia que el usuario obtuvo; este restringe el uso del sistema y brinda ciertos beneficios a quien lo utiliza, como también encontramos *software free*, el cual no requiere de una activación sino que puede ser utilizado sin ninguna restricción por parte del usuario final. | Pequeños programadores actualizando el sistema operativo de ilustración de vector plana aislada por computadora. Especialistas en TI de caricaturas actualizando software, programas y aplicaciones. Error técnico y concepto de servicio  Imagen: 228125\_i18 |
| **Desactivación** | La desactivación se realiza de dos maneras, porque depende de cómo se realizó el proceso de negociación con el usuario, es decir, si se realizó bajo la negociación de licencia y esta se ha vencido, el sistema automáticamente niega el acceso y desactiva el usuario, lo que provoca que el usuario no pueda acceder al sistema. Este proceso es el más común para evitar que un usuario utilice un sistema de información sin negociar con el dueño del aplicativo su uso, pero también se puede dar el caso de la desactivación de un sistema por causas tales como errores en los procesos o funciones que realiza; en este caso, si el sistema se encuentra en una plataforma de despliegue se puede desactivar el servicio o detener lo que provoca que el *software* no pueda ser accedido por ningún usuario del sistema, esto se utiliza para realizar mantenimiento o mejoras en el sistema como tal. | Eliminación de información privada en línea en la plantilla de la página de inicio de Internet. Botón de desactivación del botón de desactivación del carácter masculino Eliminar cuenta social. Pequeño hombre en la enorme página web de la red. Ilustración del vector de dibujos animados  Imagen: 228125\_i19 |
| **Desinstalación** | La desinstalación es un proceso que permite borrar completamente los archivos que han sido copiados en un proceso de instalación, esto implica que se borre el rastro de instalación de la aplicación dentro del dispositivo, o en el caso que se realice desde una plataforma de despliegue, borrar las configuraciones y documentos que se utilizaron para colocar en marcha inicialmente el *software*. Tenemos que tener presentes las siguientes consideraciones y es que la desinstalación tiene diferentes opciones: una es la desinstalación completa que borra completamente el rastro del aplicativo como tal y adicional a ello, elimina los datos que se hayan generado a través del aplicativo; y la otra es la eliminación del *software* pero que se conserven los archivos y datos generados. Este tipo de desinstalaciones se utilizan mucho en el caso de los *software* de videojuegos, donde se puede desinstalar el juego pero se conservan las partidas y perfiles creados del usuario para que cuando se realice nuevamente la instalación se conserven estos datos para no iniciar desde cero el proceso de manejo del sistema de información; en este caso, cuando nos referimos a la eliminación de los archivos de instalación, es borrarlos del disco duro del dispositivo, es decir, de manera permanente. En caso que se quiera utilizar nuevamente el aplicativo, se debe realizar el proceso de instalación, es decir, copiar nuevamente los archivos necesarios para que el sistema pueda ejecutarse de nuevo. | Navegador desinstalando icono Vector aislado que puede modificar o editar fácilmente  Imagen: 228125\_i20 |
| **Actualización** | La actualización es un proceso que consiste en reemplazar archivos o funciones que por algún motivo funcionan mal o han sido mejoradas de acuerdo con las versiones ofrecidas en el sistema. En algunas ocasiones hemos escuchado la expresión “lo actualicé a la versión más reciente”, lo que quiere decir es que se reemplazan los archivos por otros que contienen las nuevas funciones, en algunas ocasiones la actualización incluye archivos que no se encontraban en la versión anterior, es decir, aumenta el tamaño del sistema; por otra parte, también se puede dar el caso que algunos archivos sean obsoletos y se eliminarán para darle paso a los nuevos archivos requeridos por el sistema.  Un ejemplo de actualización es el que comentaremos a continuación. Supongamos que una compañía lanzó en 2013 un juego *online*. Quienes deseaban jugar con dicha aplicación debían tener instalado el navegador Internet Explorer. En 2015, la empresa presentó una actualización del juego que permitía disfrutarlo también en Google Chrome y en Mozilla Firefox. La actualización, por lo tanto, incrementó el alcance del juego.  Por otro lado, la actualización de los programas y demás productos digitales no siempre es algo positivo. En algunos casos, se vuelve necesario instalar nuevas versiones durante muy poco tiempo desde el momento de su lanzamiento, porque la compañía desarrolladora no ha hecho un buen trabajo optimizando su producto y necesita corregir una gran cantidad de errores para que funcione de manera correcta. | Actualización del sistema. El programador detrás de la pantalla del ordenador. Manos en el teclado. Nuevo software de versión. Instalando el proceso de actualización. Página de inicio de plantilla. Transferencia de datos. Diseño plano de ilustración de vector  Imagen: 228125\_i21 |
| **Seguimiento de versiones** | Es un proceso que se realiza para llevar una bitácora de cambios realizados en el código fuente o actualizaciones que ha tenido el sistema; adicional a ello, el control o seguimiento de versiones permite tener una copia de seguridad del proyecto o de una actualización en particular del sistema y en caso que se tengan problemas con los cambios realizados al sistema, se puede proceder a volver a la versión anterior, la cual funciona correctamente y empezar a realizar los diferentes procesos para realizar los cambios requeridos. Si vemos este proceso al detalle, es una ventaja porque permite controlar los cambios que se realizan en el proyecto de *software* y en caso que ocurra algún inconveniente, se puede regresar a la versión anterior para ver qué salió mal.  En la actualidad existen varias aplicaciones que realizan este proceso, una de ellas es GIT, la cual permite controlar las versiones del código fuente y los cambios que se realizan en el mismo- Git se ha ideado para posibilitar la ramificación y el etiquetado como procesos de primera importancia (a diferencia de SVN) y las operaciones que afectan a las ramas y las etiquetas (como la fusión o la reversión) también se almacenan en el historial de cambios. | Representación del control de versiones. Control de fuentes y proceso de desarrollo. Plantilla de infografía con múltiples pasos y ramas de desarrollo. Fases principales con burbujas de voz para la plantilla de desarrollo.  Imagen: 228125\_i22 |
| **Adaptación** | Es la capacidad que tiene un sistema de interactuar con otros sistemas y en un ambiente de trabajo de manera fluida, es decir, la adaptación le permite al sistema navegar entre versiones e infraestructura tecnológica. Este proceso, aunque en la mayoría de veces no es tan rápido como se quisiera, contempla la posibilidad de que se realice de manera progresiva. lo cual puede ser positivo o negativo dependiendo de las necesidades que se tengan desde el punto de vista del cliente.  Otros ejemplos del *software* adaptativo son aquellos que se utilizan para controlar los dispositivos externos que mejoran la experiencia de las personas con movilidad reducida. Un individuo con un uso limitado de las manos puede ser capaz de disfrutar el uso de un ordenador con la ayuda de un ratón o un *joystick* de gran tamaño y que resulta más fácil de manipular. Las personas con visión limitada se pueden beneficiar de la adición de un escáner que hace que sea posible ampliar el texto que se muestra en la pantalla o posiblemente interpretar un texto en *braille* en la voz, en este aspecto estamos viendo la adaptación desde el punto de vista de cómo el sistema puede ajustarse a las necesidades expresadas o necesarias para un cliente o usuario final esto en términos de calidad y usabilidad garantiza que el sistema muchos de estos aspectos pueden ser pensados desde los requerimientos iniciales entregados por el usuario o solo ser un valor agregado que permite la capacidad al sistema de ser operado o llegar a un mayor público. | Uso de equipos específicos aislados de ilustración de vectores de dibujos animados. Inclusión práctica escolar, libros en Braille, caracteres ampliados, dispositivos ópticos y auditivos, cartografía adaptada de vectores de software.  Imagen: 228125\_i23 |

| Tipo de recurso | Cajón de texto de color |
| --- | --- |
| Aunque sabemos que todos los productos fabricados deben tener la manera de poder retroalimentarse para poder crecer y establecerse en un mercado específico, y comprendiendo que el *software* es un producto que requiere de la integración con el usuario y que mediante esa operación puedan surgir nuevas necesidades que a su vez se conviertan en requerimientos para ser incluidas en las próximas versiones del sistema, se hace necesario que se creen los mecanismos para que el usuario participe de una manera más activa en el proceso de construcción y mejora del producto, porque esto permitirá que nuestro sistema perdure por mucho más tiempo en el mercado. | |

| Cuadro de texto |
| --- |
| Por otra parte, los roles están conectados con las responsabilidades que influyen en un proyecto de *software* y de acuerdo con la experiencia y formación de cada miembro del equipo de trabajo, aunque en algunas ocasiones una persona puede desarrollar varias funciones, es indispensable que se cuente con los conocimientos y habilidades para cumplir de manera satisfactoria este proceso. |

| **Tipo de recurso** | | Pestañas o tabs Verticales |
| --- | --- | --- |
| **Introducción** | | Conozcamos varios de estos roles y cuáles son los procesos que ejecutan: |
| Programación de la interfaz de software en dispositivos por ingenieros. Aplicación para proyecto de empresa. Un espacio de soluciones profesionales para sistemas y software. Ilustración conceptual. Vector de personas isométricas.  **Imagen:** 228125\_i24 | | |
| **Desarrolladores de aplicaciones** | El rol de desarrollador de aplicaciones es uno de los protagonistas en el desarrollo de aplicaciones informáticas porque es el encargado de realizar la codificación del sistema y dar las instrucciones necesarias para que el *software* ejecute y cumpla con todos los requisitos funcionales que se negociaron con el cliente. En algunas ocasiones también están involucrados en el proceso de despliegue y de habilitar los entornos de producción requeridos para que el sistema pueda ser visible y utilizado por los usuarios externos, que mediante el uso constante del aplicativo incluyen los datos e información necesaria para que el sistema sea operable en el tiempo. Debemos tener en cuenta que los desarrollos de las aplicaciones se realizan en diferentes lenguajes de programación, por eso se debe tener en cuenta la capacidad de adaptación del desarrollador como también su conocimiento del lenguaje y en cuáles de estos se especializa. | |
| **Gerentes de lanzamiento** | Analiza los resultados del lanzamiento de un nuevo producto. Utiliza la investigación de mercado para entender las necesidades y deseos del cliente en torno del producto. Domina todas las tareas de gestión del equipo de desarrollo de productos. Aunque muchas personas tienen la noción que este proceso no requiere de gran esfuerzo, debemos decir que es un proceso clave para un proyecto de desarrollo de *software* debido a la integración entre los diferentes clientes o usuarios del sistema y el producto desarrollado; es la persona que se involucra en este proceso de transición y convierte el producto en lo que es “un producto de calidad que brinda una solución real”. | |
| **Coordinadores de despliegue** | Este rol es el encargado de velar el proceso de pasar el sistema de un ambiente de desarrollo al ambiente de producción; esto involucra muchos procesos, entre ellos, garantizar que el nuevo ambiente proporcione las características técnicas y funcionalidades para que el sistema pueda operar de manera eficiente y brinde al usuario final una experiencia agradable y deseada. Este proceso debe ser ejecutado con mucho cuidado, puesto que en algunas ocasiones es la cara visible del proyecto, es decir, es donde toda la empresa y el cliente están pendientes a conocer realmente la operación del sistema y si este proceso llega a fallar, el producto pierde credibilidad; por ello es indispensable que el coordinador de despliegue se asegure que este proceso quede realizado de la mejor manera. | |
| **Administrador de sistema** | El administrador del sistema es la persona encargada de realizar los procesos de soporte y mantenimiento al aplicativo una vez este ha sido desplegado y se encuentra en el ambiente de producción; es importante destacar las competencias del administrador del sistema, ya que es capaz de dar soluciones a los problemas que se puedan presentar luego de ser implementada la solución todos estos reportes serán tenidos en cuenta en caso que se requiera realizar ciertos cambios en el sistema o fallos en su operación; adicional a ello también es responsable de la creación de usuarios y datos para que la operación del sistema sea posible. | |
| **Administrador de base de datos** | El administrador de base de datos cumple una función primordial en la administración de los datos y seguridad del aplicativo desarrollado, debido a que es el encargado de velar porque la información que se almacena cumpla con los requerimientos de seguridad y disponibilidad; esto hace referencia a la importancia de resguardar los datos producidos a través del aplicativo, de tal manera que no se corrompan o se pierdan. Otro proceso fundamental que lidera el administrador de la base de datos es determinar los permisos de ejecución y de acceso a la información por parte de los usuarios que utilizan el sistema, debe definir permisos y perfiles de usuario de acuerdo con las restricciones que cada uno de ellos pueda presentar. | |
| **Coordinadores de liberación** | La liberación del *software,* podríamos decir de alguna manera, es la fase final del proceso de entrega del producto final al usuario. Antes de realizar este proceso se debe garantizar que todos los procesos de pruebas, despliegue y adaptación del nuevo sistema al ambiente de producción se han realizado de manera correcta, esto implica también la aplicación de las listas de chequeo para verificar que todos los pasos y procedimientos se realizaron de manera exitosa. Este rol es el encargado de la entrega oficial del sistema desarrollado al usuario final, dando por hecho que aquello en lo cual se sentaron un día a negociar y que fue la base de los requerimientos exigidos por el usuario, se ha cumplido en buenos términos, de allí se firman los documentos oficiales de la entrega del producto totalmente terminado y con puesta en marcha. | |

| Cuadro de texto |
| --- |
| Finalizamos el estudio del componente formativo con los tipos de despliegue, los cuales muestran la manera como un *software* pasa a producción y de acuerdo con las necesidades tecnológicas del usuario final, se debe determinar de qué manera se realizará este proceso, aunque existen algunos que pueden variar de acuerdo con las formas de negociación que se establezcan con el usuario. |

| **Tipo de recurso** | Slider Presentación | |
| --- | --- | --- |
| **Introducción** | A continuación, veremos tres tipos de despliegues para proyectos de *software*: | |
| **Autónomo**  Un despliegue autónomo incluye la base de datos, la consola de gestión y un agente desplegado en un único *host*. Este despliegue se utiliza normalmente para fines de evaluación y desarrollo, los cuales suelen realizarse para la comprobación del funcionamiento del aplicativo; para este caso particular, solo se cuenta con un *host* y la consola de administración, lo que quiere decir que es un sistema básico de despliegue y que solo incluirá una dependencia del sistema, la cual será administrada desde consola y no cuenta con una interfaz de administración que permita realizar este proceso de una manera más sencilla. Se requieren conocimientos técnicos de comandos y rutinas para realizar este tipo de procesos; ahora bien, se comprende también este proceso como la manera más simple de realizar el despliegue de un sistema de información en un ambiente de producción; recordemos que se coloca a disposición del usuario final para que pueda hacer uso del mismo de acuerdo con las negociaciones de acceso, es decir, por ejemplo una VPN o Internet de acuerdo con las políticas de acceso que se determinen. Es importante agregar que este tipo de despliegues suelen ser sencillos en el sentido de que los archivos, tales como bases de datos y de código fuente, se suben de manera rápida al no tener la carga de un entorno de administración complejo. | | Programador de desarrollo profesional trabajando en la programación de un sitio web de software y tecnología de codificación, escribir códigos y código de datos, Programación con HTML, PHP y javascript.  **Imagen:** 228125\_i25 |
| **Instalación de producción normal**  Una instalación normal coloca la base de datos, la consola de gestión y los agentes en distintos *hosts*. Este despliegue es el entorno de producción utilizado habitualmente, para estos casos los *host* o sistemas de soporte de la instalación donde corre la aplicación pueden ser muchos, los cuales permiten generar un balance en la carga y eficiencia del sistema. Esto en términos de rendimiento y velocidad de respuesta es mucho más eficiente y el usuario final tendrá una mejor experiencia en su uso, otro aspecto a tener en cuenta es que, al separar cada componente como base de datos, consola de gestión y agentes en distintos *hosts,* se tiene un mayor nivel de seguridad. | | Desarrollo de tecnologías de programación y codificación. Diseño del sitio web. Concepto de ciberespacio.  **Imagen:** 228125\_i26 |
| **Instalación escalonada**  En un despliegue escalonado, normalmente hay tres despliegues normales:   1. Desarrollo (también denominado *scratch* o *sandbox*): el personal de desarrollo lo utiliza para crear proyectos nuevos. También se utiliza como el área inicial para las actualizaciones. 2. Prueba: el personal de desarrollo y asesoramiento de calidad lo utilizan para probar el sistema. 3. Producción: lo utiliza todo el personal para ejecutar trabajos en operaciones de negocio cotidianas.   Esto es muy importante tenerlo en cuenta, porque al pasar por todos los procesos, desde ambientes de desarrollo hasta producción, se puede visualizar el comportamiento que está teniendo en los mismos y si existe algún problema en alguna de esas etapas se puede corregir antes de que cause algún inconveniente mayor. | | Retrato de desarrollador afroamericano que usa laptop para escribir código sentado en un escritorio con múltiples pantallas analizando algoritmo en agencia de software. Coder que trabaja en la interfaz de usuario mediante un equipo portátil.  **Imagen:** 228125\_i27 |

| Cuadro de texto |
| --- |
| Recuerde explorar los demás recursos que se encuentran disponibles en este componente formativo; para ello, diríjase al menú principal donde encontrará la síntesis, una actividad didáctica para reforzar los conceptos estudiados, material complementario, entre otros. |

**SÍNTESIS**

| **Tipo de recurso** | Síntesis |
| --- | --- |
| Desarrollo de aplicaciones web full stack  Síntesis: Despliegue de *software.* | |
| **Introducción** | El siguiente mapa integra los criterios y especificidades de los conocimientos expuestos en el presente componente formativo. |
|  | |

**ACTIVIDAD DIDÁCTICA**

| **Tipo de recurso** | Actividad didáctica. Arrastrar y soltar | |
| --- | --- | --- |
| Esta actividad le permitirá determinar el grado de apropiación de los contenidos del componente formativo.  De acuerdo con la definición planteada en la columna izquierda, arrastre cada término al lugar que considere correcto de la columna derecha. | | **Imagen:** 228125\_i29 |
| Plataforma implementada por la empresa Microsoft para el despliegue de aplicaciones de cualquier tipo. | | **AZURE** |
| Plataforma implementada por la empresa Amazon para el despliegue de aplicaciones web y bases de datos. | | **AWS** |
| Es una herramienta de integración continua de código fuente en la cual se puede realizar seguimiento y depuración del mismo. | | **JENKINS** |
| Es una herramienta de repositorio de código la cual permite visualizar los cambios realizados en el mismo y sus versiones. | | **GITLAB** |
| Se encuentran ligados a las responsabilidades y funciones que ejecuta cada miembro del equipo de trabajo. | | **Roles** |
| Es el encargado de velar por el correcto diseño y almacenamiento de la información que genera el *software*. | | **Administrador de bases de datos** |

**MATERIAL COMPLEMENTARIO**

| Tipo de recurso | Material complementario | | |
| --- | --- | --- | --- |
| Tema | Referencia APA del material | tipo | Enlace |
| Herramientas de despliegue | Païola, P. (2021). *Microsoft Azure : Gestione su Sistema de Información en la Nube*. Ediciones ENI. <https://sena-primo.hosted.exlibrisgroup.com/permalink/f/1j5choe/sena_biblioteca_eniEPT3GIT> | Libro | https://sena-primo.hosted.exlibrisgroup.com/permalink/f/1j5choe/sena\_biblioteca\_eniEPT3GIT |
| Herramientas de despliegue | Rojas-Albarracin, G. et al. (2017). Plataforma computacional sobre Amazon Web Services de renderizado distribuido/Computational Platform About Amazon Web Services Distributed Rendering/Plataforma de computacao sob Amazon Web Services AWS de renderizacao distribuida. *Revista Cientifica Del Centro De Investigaciones Y Desarrollo Cientifico De La Universidad Distral Francisco Jose De Caldas,* (30). <https://sena-primo.hosted.exlibrisgroup.com/permalink/f/1i756fj/TN_cdi_gale_infotracmisc_A510482190> | Artículo | <https://sena-primo.hosted.exlibrisgroup.com/permalink/f/1i756fj/TN_cdi_gale_infotracmisc_A510482190> |
| Ambientes de producción o desarrollo | Sarosi, A. (2022). *Entorno de Desarrollo vs Producción: El proceso de publicar una App Web* (video). YouTube. <https://www.youtube.com/watch?v=9GaVCsPuDB4> | Video | <https://www.youtube.com/watch?v=9GaVCsPuDB4> |
| Actividades, roles y tipos de despliegue | DesarrolloWeb.com. (2020). *Hosting y despliegue continuo gratuito con GitHub y Netlify* (video). YouTube. <https://www.youtube.com/watch?v=OomyaFBGlyU> | Video | <https://www.youtube.com/watch?v=OomyaFBGlyU> |

**GLOSARIO**

| **Tipo de recurso** | Glosario |
| --- | --- |
| AZURE: | plataforma soportada por Microsoft para el despliegue de aplicaciones informáticas. |
| Actualización: | proceso que reemplaza o copia archivos nuevos en el disco duro de una computadora o servidor. |
| Ambiente de desarrollo: | espacio donde se desarrolla el sistema de manera local y se realizan las pruebas iniciales del sistema. |
| Ambiente de producción: | espacio donde se coloca en marcha el *software* desarrollado para que los usuarios finales puedan usarlo. |
| AWS: | plataforma soportada por Amazon que presta los servicios de despliegue de aplicaciones informáticas. |
| Desinstalación: | proceso de eliminar los archivos de instalación de una aplicación del disco duro de un dispositivo. |
| GIT: | aplicación que permite llevar un control de código fuente de un aplicativo que se desarrolla. |
| Instalación: | proceso de copiar los archivos necesarios para poner en marcha un sistema de información. |
| JENKINS: | sistema para realizar el proceso de integración continua y pruebas de *software*. |
| Reporte: | consulta que integra varios datos e información de un proceso y se realiza de manera organizada. |

**REFERENCIAS BIBLIOGRÁFICAS:**

| **Tipo de recurso** | Bibliografía |
| --- | --- |
| IBM Documentación. (2022). *Entornos de desarrollo y producción.* <https://www.ibm.com/docs/es/baw/20.x?topic=overview-case-development-production-environments> | |
| Microsoft Soporte. (s.f.). *Access SQL: conceptos básicos, vocabulario y sintaxis.* <https://support.microsoft.com/es-es/office/access-sql-conceptos-b%C3%A1sicos-vocabulario-y-sintaxis-444d0303-cde1-424e-9a74-e8dc3e460671#bm2> | |
| NetApp. (s.f.). *¿Qué es DevOps?* <https://www.netapp.com/es/devops-solutions/what-is-devops/> | |
| Ortiz Clavijo, L. F. et al. (2018). Computación En La Nube: Estudio De Herramientas Orientadas a La Industria 4.0. *Lampsakos,* (20), p. 68-75. <https://revistas.ucatolicaluisamigo.edu.co/index.php/lampsakos/article/view/2560> | |
| Quezada Sarmiento, P. y Suárez-Guerrero, C. (2021). La Computación en la Nube en el proceso formativo en Programación Web. RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao, (E42), p. 1-11. <https://www.researchgate.net/publication/346556176_La_Computacion_en_la_Nube_en_el_proceso_formativo_en_Programacion_Web> | |
| Turingears. (s.f.). *¿Qué es GitLab?* <https://turingears.com/que-es-gitlab/> | |