PLAN DE GESTIÓN DE LA CONFIGURACIÓN

Bugfund

versión 2.0

Lima, 28 de abril del 2023

Control de Versiones

| Fecha | Versión | Descripción | Autor |
| --- | --- | --- | --- |
| 19/04/2023 | 1.0 | Creación y Redacción del Documento | Todos |
| 20/04/2023 | 1.1 | Modificación y Edición del Documento | Todos |
| 28/04/2023 | 1.2 | Modificación y Edición del Documento | Ayrton Lopez/Ademir Vásquez |
| 07/06/2023 | 2.0 | Modificación y Edición del Documento | Ayrton Lopez |

1. **Introducción**
   1. Situación de la empresa

Bugfund es una empresa creada por estudiantes de la Universidad Nacional Mayor de San Marcos, los cuales poseen experiencia en el mercado dedicado a desarrollar y brindar soluciones informáticas. Actualmente Bugfund cuenta con cuatro proyectos en mantenimiento y uno en desarrollo. Los cuatro proyectos en mantenimiento son proyectos desarrollados previamente por Bugfund como el sistema gestor de venta de pequeñas minimarkets, un sistema de toma de asistencia con reconocimiento facial, un software para la detección de billetes falsos y un sistema para sacar reserva en comedores locales, actualmente está en desarrollo un sistema para llevar la cuenta de la dieta y actividad física para personas en rehabilitación. Los problemas que enfrenta Bugfund son:

* Conflictos de versión: los miembros del equipo de desarrollo trabajan en diferentes versiones del mismo código, lo que resulta en conflictos de integración y en la necesidad de fusionar el código manualmente.
* Pérdida de código fuente: el código fuente de un proyecto importante se pierde debido a una mala gestión de la copia de seguridad, lo que resulta en una gran cantidad de trabajo de reconstrucción.
* Dificultades de colaboración: los miembros del equipo de desarrollo tienen dificultades para colaborar y coordinar sus esfuerzos, lo que resulta en un código mal estructurado y con errores.
* Incapacidad para rastrear errores: la empresa no tiene una manera eficaz de rastrear y documentar los errores, lo que hace que sea difícil identificar y resolver los problemas.
  1. Propósito de Implementar la Gestión de la Configuración del Software

El propósito de implementar la Gestión de la configuración del Software es asegurar que el software (programa) desarrollado cumpla con los requerimientos del cliente y esté bajo control en todo momento durante el ciclo de vida del desarrollo de software. La gestión de la configuración del software se enfoca en identificar, controlar y gestionar los elementos del software, incluyendo su documentación asociada, a lo largo del proceso de desarrollo y mantenimiento del software.

Entre los beneficios de implementar la Gestión de la Configuración del Software se incluyen:

* Control de cambios: la gestión de configuración del software permite controlar los cambios realizados en el software, asegurando que se registren y se implementen de manera controlada.
* Gestión de versiones: la gestión de configuración del software permite gestionar las diferentes versiones del software, lo que facilita el seguimiento de los cambios y ayuda a asegurar que el software esté siempre bajo control.
* Facilita la colaboración: al establecer una estructura para la gestión de la configuración del software, se facilita la colaboración entre los miembros del equipo de desarrollo.
* Asegura la calidad: la gestión de la configuración del software ayuda a garantizar la calidad del software, ya que permite realizar pruebas y verificaciones de manera controlada.
* Facilita el mantenimiento: la gestión de la configuración del software permite rastrear los cambios realizados en el software, lo que facilita el mantenimiento y la solución de problemas.
  1. Benchmarking

Tabla 1

*Tabla de Benchmarking de otras herramientas*

| Criterios | Git | TFS | Mercurial | SVN |
| --- | --- | --- | --- | --- |
| Licencia | Open Source GPL v2 | Propietaria | Open Source GPL v2 | Open Source |
| Integración | Amplia integración | Integración completa | Amplia integración | Limitada integración |
| Rendimiento | Alto rendimiento | Alto rendimiento | Rendimiento moderado | Rendimiento moderado |
| Escalabilidad | Escalable a gran escala | Escalable a gran escala | Escalable a gran escala | Escalable a mediana escala |
| Usabilidad | Interfaz de línea de comandos | Interfaz de línea de comandos, Interfaz gráfica de usuario | Interfaz gráfica de usuario | Interfaz de línea de comandos |
| Ramas y etiquetas | Fácil manejo de ramas y etiquetas | Fácil manejo de ramas y etiquetas | Fácil manejo de ramas y etiquetas | Manejo moderado |
| Soporte | Gran comunidad | Soporte técnico de pago | Soporte técnico de pago | Comunidad y soporte moderados |
| Costo | Gratuito | Pago | Gratuito | Gratuito |

*Nota.* Autoría propia.

* 1. Herramienta elegida
     1. Introducción

Team Foundation Server (TFS) es la plataforma de colaboración en el centro de la solución de administración del ciclo de vida de las aplicaciones ALM( Administración del ciclo de vida de la aplicación) de Microsoft.

Proporciona un conjunto de herramientas de desarrollo de software de colaboración que se integran con su editor o IDE existente, lo que permite a su equipo interdisciplinario trabajar de manera eficiente en proyectos de software de todos los tamaños.

Permite capacidades de DevOps que abarcan todo el ciclo de vida de la aplicación. Los principales elementos incluyen la gestión del código fuente, la gestión de requisitos, la gestión de proyectos, la generación de informes, las pruebas y las capacidades de gestión de versiones.

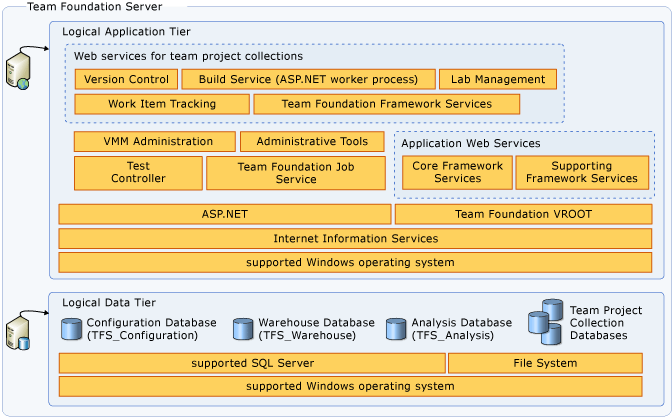
* + 1. Arquitectura

La arquitectura de TFS se basa en un modelo cliente-servidor, donde los clientes pueden acceder al servidor de TFS para acceder a los diferentes servicios y recursos que proporciona. TFS consta de varios componentes, entre ellos:

* + - * Servidor de aplicaciones: es el componente principal de TFS y aloja los servicios de la aplicación, como el servicio de control de versiones, el servicio de gestión de proyectos y el servicio de informes. El servidor de aplicaciones se ejecuta en Microsoft Internet Information Services (IIS).
      * Base de datos: es el componente que almacena los datos del sistema, como los metadatos del proyecto, el código fuente y los datos de seguimiento de errores. TFS utiliza Microsoft SQL Server para alojar la base de datos.
      * Cliente de escritorio: es el componente que se utiliza para acceder a TFS desde un equipo de escritorio. Proporciona una interfaz de usuario gráfica para acceder a los diferentes servicios y recursos de TFS.

Figura 1

*Arquitectura de TFS*



Aaronhallberg. (2022, 24 octubre). Architecture - Azure DevOps. Microsoft Learn. https://learn.microsoft.com/en-us/azure/devops/server/architecture/architecture?view=azure-devops-2022

* + 1. Beneficios
       - Gestión de proyectos: Ayuda a que el desarrollo de software sea más veloz y más fácil. Puedes usarlo desde el inicio del proyecto hasta la creación y eventualmente hasta la implementación.
       - Repositorio de código: Tiene repositorios de código limitados, puede almacenar todos sus códigos en un solo software. Colabora con el resto del equipo y configura quién puede obtener acceso a diferentes repositorios.
       - Consistencia de datos: Garantiza que pueda obtener la migración de datos sin dificultades. Puede transferir sus datos y aplicaciones existentes de forma segura y sin miedo.
       - Implementación Híbrida: Se puede implementar en Cloud, Edge y On-premise. Se puede elegir si desea que sus datos se almacenen en un servidor remoto y acceder a ellos utilizando cualquier dispositivo.
       - Control de calidad: Le ayuda a verificar posibles problemas mientras desarrolla el software para que se pueda ahorrar tiempo y al mismo tiempo producir aplicaciones de calidad.

1. **Roles**

**Tabla 2**

*Miembros del grupo asignados a su rol*

| Rol | Persona asignada | Responsabilidad |
| --- | --- | --- |
| Gestor de la configuración | Lopez Ticona, Ayrton Edgar Santiago | Encargado de implementar, mantener y mejorar la Gestión de la Configuración. |
| Bibliotecario | Estrada López, Josué Augusto | Define y da mantenimiento a las bibliotecas que se usan durante la Gestión de la Configuración. |
| Miembros del equipo de proyecto | Calixto Alejo, Alan Franco Albeyro  Valentín Mejía, Jorge Luis  Vásquez Marcani, Ademir Raúl  Ostos Torres, Deysi Analí  Valle Atencio, Luis Alfonso | Consultan y están al tanto de la información de la Gestión de la Configuración. |

*Nota.* Autoría propia.

1. **Actividades de la SCM**
   1. Identificación
      1. Definición de nomenclaturas de los ítems

| Caso 1:  Para ítems que son específicos de un proyecto  Acrónimo del proyecto + “-” + Acrónimo del ítem + extensión |
| --- |

| Caso 2:  Para ítems que no pertenecen a un proyecto en específico  Acrónimo del ítem + extensión |
| --- |

| Caso 3:  Para diferenciar ítems de un proyecto con un mismo acrónimo  Acrónimo del proyecto + “-” + Acrónimo del ítem + Siguiente letra que los diferencia + extensión |
| --- |

* + 1. Inventario de ítems

A continuación, se mostrará una tabla la cual muestra información de los ítems que se están trabajando. Se debe tener en cuenta el tipo al que pertenecen (E = Evolución, F = Fuente y S = Soporte).

**Tabla 3**

*Inventario de ítems*

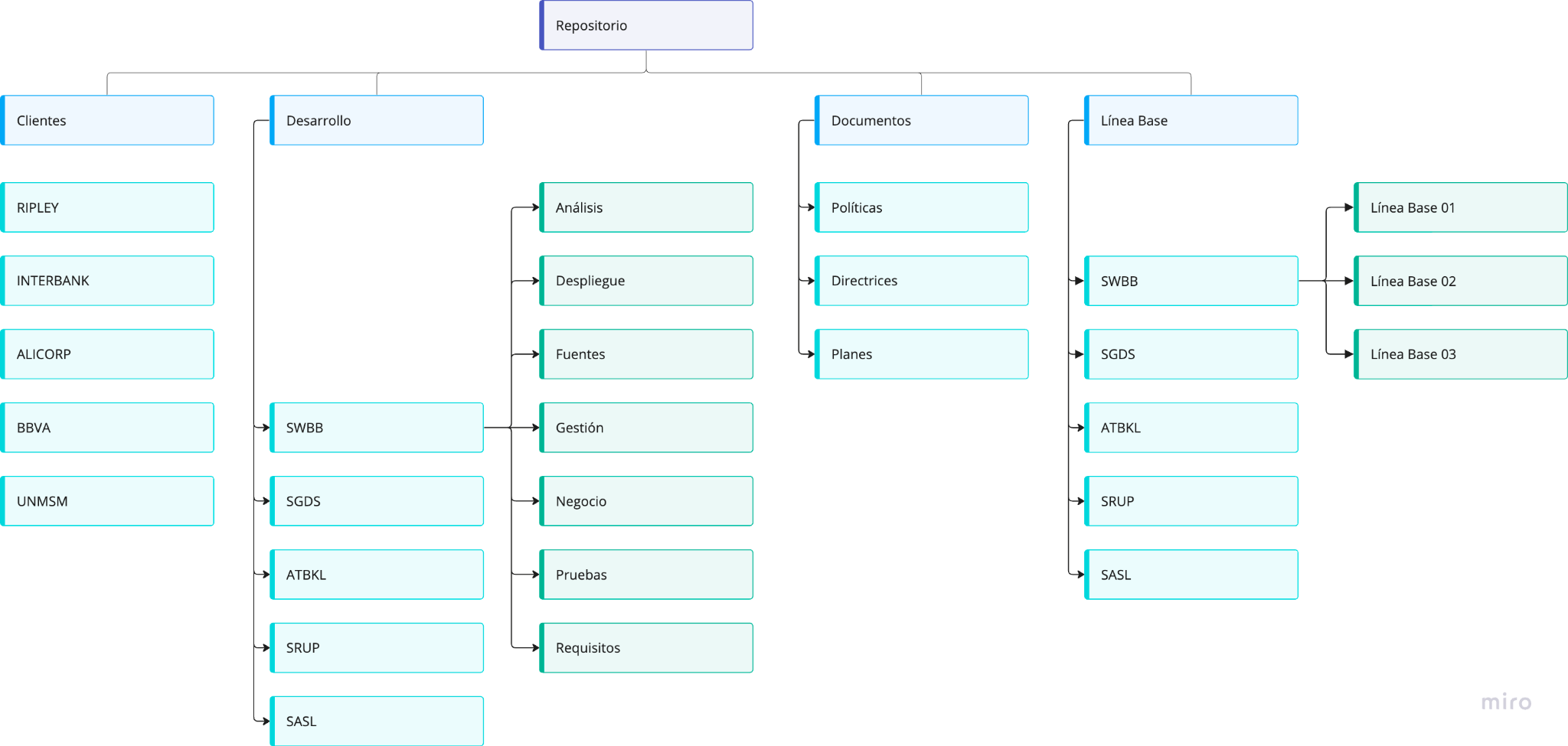
| Tipo | Nombre del Ítem | Nomenclatura | Extensión | Proyecto |
| --- | --- | --- | --- | --- |
| S | Visual Studio Code | VSC | exe | SWBB |
| S | JetBrains DataGrip | DG | exe | SWBB |
| S | MySQL Workbench | MSQLWB | exe | SWBB |
| S | Figma | F | exe | SWBB |
| E | Plan de la Gestión de la Configuración | PGC | docx | SWBB |
| E | Project Charter | SWBB-PC | docx | SWBB |
| E | Cronograma del Proyecto | SWBB-CP | docx | SWBB |
| E | Documento de Requisitos de Usuario | SWBB-DRU | docx | SWBB |
| E | Documento(s) de Especificación de Requisitos | SWBB-DER | docx | SWBB |
| E | Documento de Arquitectura de Software | SWBB-DAS | docx | SWBB |
| E | Documento de Diseño de base de Datos | SWBB-DDBD | docx | SWBB |
| E | Documento de Wireframes | SWBB-DW | docx | SWBB |
| E | Documento de Guía de Estilos | SWBB-DGE | docx | SWBB |
| F | Módulo Crear usuario | SWBB-MCY | js | SWBB |
| F | Módulo Buscar barbero | SWBB-MBB | js | SWBB |
| F | Módulo Gestionar información personal | SWBB-MGIP | js | SWBB |
| F | Módulo Gestionar servicios privados | SWBB-MGSP | js | SWBB |
| F | Módulo Gestionar reservas | SWBB-MGR | js | SWBB |
| F | Módulo Gestionar fotos | SWBB-MGF | js | SWBB |
| F | Módulo Visualizar reservas programadas | SWBB-MVRP | js | SWBB |
| F | Script de la Base de Datos | SWBB-BD | sql | SWBB |
| F | Documento de Pruebas Unitarias | SWBB-DPU | docx | SWBB |
| F | Documento de Desarrollo del Software | SWBB-DDS | docx | SWBB |
| F | Documento de Sprint Retrospective | SWBB-DSR | docx | SWBB |
| F | Acta de Cierre del Proyecto | SWBB-ACP | docx | SWBB |

*Nota.* Autoría propia.

* + 1. Diseño de la estructura del repositorio

**Figura 2**

*Estructura del repositorio*



*Nota.* Autoría propia.

* + - * Línea Base

El responsable de mantener actualizadas las tres líneas bases (Línea base 01, línea base 02 y Línea base 03) es el Gerente de Configuración. Los permisos según los roles se muestran a continuación.

**Tabla 4**

*Roles y tipos de acceso*

| Rol | Tipo de acceso |
| --- | --- |
| Gerente de Configuración | Leer, escribir, ejecutar y eliminar. |
| Gerente de Proyecto | Leer, escribir, ejecutar y eliminar. |
| Desarrolladores | Leer y ejecutar. |
| Bibliotecario | Leer, escribir, ejecutar y eliminar. |

*Nota.* Autoría propia.

* + - * Documentos

El responsable de planificar las distintas fases del proyecto y de supervisar la documentación (Directrices, planes y políticas) a realizar es el Gerente de Proyecto.

**Tabla 5**

*Roles y tipos de acceso*

| Rol | Tipo de acceso |
| --- | --- |
| Gerente de Proyecto | Leer, escribir, ejecutar y eliminar. |
| Desarrolladores | Leer y ejecutar. |
| Bibliotecario | Leer, escribir, ejecutar y eliminar. |

*Nota.* Autoría propia.

* + - * Clientes

Esta biblioteca contiene los clientes con los que se ha y/o está trabajando, tales como ALICORP, BBVA, INTERBANK y RIPLEY. El Gerente del Proyecto es el responsable de actuar como interlocutor ante promotores, accionistas y también ante los equipos de trabajo y participantes del proyecto.

**Tabla 6**

*Roles y tipos de acceso*

| Rol | Tipo de acceso |
| --- | --- |
| Gerente del proyecto | Leer, escribir, ejecutar y eliminar. |
| Bibliotecario | Leer, escribir, ejecutar y eliminar. |

*Nota.* Autoría propia.

* + - * Desarrollo

Los encargados de producir código limpio y eficiente son los desarrolladores. En esta biblioteca se encuentran los proyectos (ATBKL, SASL, SGDS, SRUP y SWBB) que se están realizando y los que se han realizado con nuestros clientes.

**Tabla 7**

*Roles y tipos de acceso*

| Rol | Tipo de acceso |
| --- | --- |
| Desarrolladores | Leer, escribir, ejecutar y eliminar. |
| Tester | Leer y ejecutar. |
| Bibliotecario | Leer, escribir, ejecutar y eliminar. |

*Nota.* Autoría propia.

* + 1. Definición de línea base
       - Proyecto Sistema Web de Booking para Barberos

**Tabla 8**

*Líneas bases y sus ítems*

| Nombre de la línea base | Ítems |
| --- | --- |
| Línea Base 01 | Project Charter.  Lista de Historia de Usuarios.  Documentos de Especificación de Requisitos (1, 2, 3, 4, ,5 ,6 y 7).  Documento de Arquitectura de Software.  Documento de Diseño de Base de Datos.  Documento de Wireframes.  Documento de Guía de Estilos.  Documento de Sprint Retrospective. |
| Línea Base 02 | Implementación de la Base de Datos.  Implementación de la funcionalidad Crear usuario.  Implementación de la funcionalidad Buscar barbero.  Implementación de la funcionalidad Gestionar información personal.  Implementación de la funcionalidad Gestionar servicios.  Documento del Desarrollo del Software.  Documento de Sprint Retrospective. |
| Línea Base 03 | Script de datos de la Base de Datos.  Documento de Pruebas Unitarias.  Implementación de la funcionalidad Gestionar reservas.  Implementación de la funcionalidad Gestionar fotos.  Implementación de la funcionalidad Visualizar reservas programadas.  Documento del Desarrollo del Software.  Documento de Sprint Retrospective.  Acta de Cierre del Proyecto. |

*Nota.* Autoría propia.

* 1. Control de la Configuración
     1. Ciclo de vida de ítems

Clasificación de ítems del proyecto SWBB que se encuentran en las etapas Borrador y Aceptado, siendo hoy 7 de junio del 2023.

**Tabla 9**

*Fechas de Inicio y final de cada ítem*

| Ítem | Etapa | F. Inicio | F. Final |
| --- | --- | --- | --- |
| Plan de la Gestión de la Configuración | A | docx | 16/04/2023 |
| Project Charter | A | 11/04/2023 | 16/04/2023 |
| Cronograma del Proyecto | A | 11/04/2023 | 16/04/2023 |
| Documento de Requisitos de Usuario | A | 12/04/2023 | 15/04/2023 |
| Documento(s) de Especificación de Requisitos | A | 16/04/2023 | 28/04/2023 |
| Documento de Arquitectura de Software | A | 16/04/2023 | 29/04/2023 |
| Documento de Diseño de base de Datos | A | 16/04/2023 | 29/04/2023 |
| Documento de Wireframes | A | 16/04/2023 | 29/04/2023 |
| Documento de Guía de Estilos | A | 16/04/2023 | 29/04/2023 |
| Módulo Crear usuario | B | 25/05/2023 | 11/06/2023 |
| Módulo Buscar barbero | B | 25/05/2023 | 11/06/2023 |
| Módulo Gestionar información personal | B | 25/05/2023 | 11/06/2023 |
| Módulo Gestionar servicios privados | B | 25/05/2023 | 11/06/2023 |
| Módulo Gestionar reservas | B | 25/05/2023 | 19/06/2023 |
| Módulo Gestionar fotos | B | 25/05/2023 | 19/06/2023 |
| Módulo Visualizar reservas programadas | B | 25/05/2023 | 19/06/2023 |
| Script de la Base de Datos | B | 25/05/2023 | 09/06/2023 |
| Documento de Pruebas Unitarias | B | 11/06/2023 | 20/06/2023 |
| Documento de Desarrollo del Software | B | 20/06/2023 | 21/06/2023 |
| Documento de Sprint Retrospective | B | 20/06/2023 | 21/06/2023 |
| Acta de Cierre del Proyecto | B | 16/06/2023 | 21/06/2023 |

*Nota.* Autoría propia.

1. Referencias

* Microsoft. (s.f.). Plan and track work with backlogs and boards. Team Foundation Server. https://docs.microsoft.com/en-us/azure/devops/boards/backlogs/?view=azure-devops
* Microsoft. (s.f.). Manage your code in Git. Team Foundation Server. https://docs.microsoft.com/en-us/azure/devops/repos/git/?view=azure-devops
* Microsoft. (s.f.). Overview of Team Foundation Server. Team Foundation Server.https://docs.microsoft.com/en-us/azure/devops/server/overview/?view=azure-devops
* Para la definición de arquitectura de software: Bass, L., Clements, P., & Kazman, R. (2012). Software Architecture in Practice (3rd ed.). Addison-Wesley Professional.