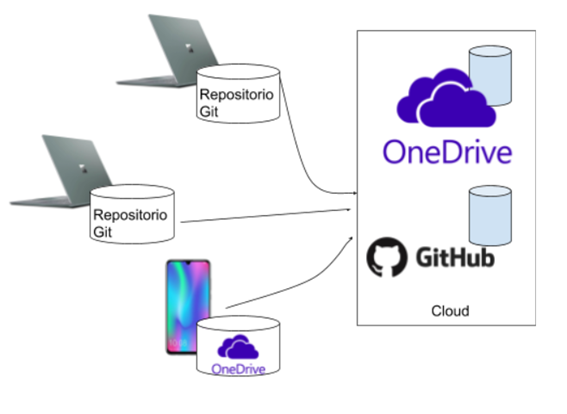
**Herramientas, entorno e infraestructura**

Se decidió utilizar herramientas e infraestructura en la nube para la gestión de la configuración de software y sus entregables, las herramientas utilizadas fueron las siguientes:

* GitHub: GitHub es un sitio web y un servicio en la nube que ayuda a los desarrolladores a almacenar y administrar su código, al igual que llevar un registro y control de cualquier cambio sobre este código. Ofrecen sus servicios de distintas maneras:
  + Uso individualizado: Es de uso gratuito, pero también posee una versión “Pro” que ofrece mayores funcionalidades por 7 dólares al mes.
  + Uso entre equipo: Cuesta 9 dólares al mes y posee diferentes funcionalidades como control del equipo, manejo de las cuentas y repositorios.
* Microsoft OneDrive: OneDrive es el servicio en la nube de Microsoft que le conecta a todos los archivos. Le permite almacenar y proteger sus archivos, compartirlos con otros usuarios y obtener acceso a ellos desde cualquier lugar de todos los dispositivos. Ofrece sus servicios para diferentes tipos de entidades:
  + Personal: Ofrece su alojamiento en la nube de manera gratuita pero con solo 5GB para ser usado, también posee otras tarifas desde 70 dólares al año donde te viene, junto al alojamiento en la nube, office 365.
  + Empresa: Para empresas tiene la función de usar los productos office con otros usuarios en simultáneo. Su tarifa va desde 5 dólares por usuario al mes hasta 12.5 por usuario al mes

**Dibujo 1. Arquitectura de la SCM**

**Lista de la clasificación de los ítems de la configuración**

|  |  |  |  |
| --- | --- | --- | --- |
| Tipo de ítem (Evolución = E, Fuente = F, Soporte = S) | Nombre de item | Origen | Proyecto |
| S | Visual Studio Code-1.38.1 | Proveedor | SGE |
| S | PyCharm-2019.2.3 | Proveedor | SVO |
| S | IntelliJ IDEA-2019.2.3 | Proveedor | SAA |
| S | Android Studio-3.5.1 | Proveedor | SEM |
| S | Windows 10 | Proveedor | - |
| S | Git-2.23.0 | Proveedor | - |
| S | Gradle 3.1 | Proveedor | SEM |
| S | Google Analytics | Proveedor | SGE |

**Definición de la nomenclatura de ítem**

* Caso 1:

|  |
| --- |
| Ítems que no son específicos a un proyecto único, tales como políticas, descripciones de procesos y guías, son identificados únicamente por su acrónimo.  Ejemplo: PGC (Plan de Gestión de la Configuración). |

* Caso 2:

|  |
| --- |
| Los ítems que son específicos de un proyecto utilizan un identificador de dos partes: ACRÓNIMO PROYECTO y ACRÓNIMO derivado del tipo de artefacto.  Ejemplo, para identificar el plan de gestión de la configuración del proyecto SGE, tenemos: SGE-PP. |

* Caso 3:

|  |
| --- |
| Los ítems que son de tipo DECUS de un proyecto utilizan un identificador de tres partes: ACRÓNIMO PROYECTO, ACRÓNIMO derivado del tipo de artefacto y número en orden ascendente con 3 dígitos.  Ejemplo: SGE-DECUS-001. |

* Caso 4:

|  |
| --- |
| Nivel de versión   * El nivel de versión de cada ítem se mantiene como un identificador separado. Esto permite que el identificador principal sea utilizado como parte del nombre del ítem, para acceder a la versión más actualizada sin necesidad de requerir cambios a todos los ítems referenciados. * El nivel de versión se mantiene como un identificador numérico con dos componentes: Versión. Revisión. Ejemplo: 1.1; Versión 1, Revisión 1. |

* Caso 5:

|  |
| --- |
| Número de versión   * El número de versión cambia después de ser evaluado, cuando ya se excede la versión # .9, o cuando el ítem es completamente reconstruido. En este caso la versión 1. # se convertirá en versión 2.0. |

* Caso 6:

|  |
| --- |
| Número de revisión   * El número de revisión cambia cuando el contenido ha cambiado, pero la estructura principal y el flujo del ítem se mantiene igual. La secuencia normal de las revisiones es: 1.0, 1.1, 1.2, etc. |