1. **Introducción**
   1. **Propósito**

El siguiente documento proporciona una descripción de la arquitectura del sistema que consiste en un sitio web de una Bolsa de Empleos. Con el fin de dar una vista en todos los niveles para que así el desarrollo y modificación del mismo sea favorable para todas las partes involucradas.

* 1. **Alcance**

La elaboración de este documento permite considerar los factores involucrados en el desarrollo del software, El documento contiene visión general mediante el diseño de diagramas dado el análisis previo de los requisitos del sistema para darle solución a una problemática presentado por un cliente. Además, que Forma la columna vertebral para construir un software y es en gran medida responsable de permitir o no ciertos atributos de calidad del sistema.

* 1. **Definiciones, acrónimos y abreviaciones**

**UML**: El lenguaje unificado de modelado (UML, por sus siglas en inglés, Unified Modeling Language).

**GANTT**: Nombrado así por Henry Laurence Gantt quien, entre 1910 y 1915, modificó y divulgó este tipo de diagrama en occidente.

**VISTAS**: Es una representación de un área de interés o perspectiva del sistema en alto nivel.

**TIPOS DE VISTAS**: Especificación de una convención de cómo construir y usar una vista. Deben satisfacer la capacidad de creación y análisis de una vista.

1. **Representación Arquitectónica**
   1. **Hechos más Importantes**

Es un sitio web que se utilizara para la búsqueda y publicación de puesto de trabajo. Los usuarios podrán postularse y aplicar para los distintos empleos publicados en la plataforma. El sitio contará con un inicio de sesión para que los usuarios puedan hacer sus búsquedas a partir de su perfil.

A continuación, veremos una serie de diagramas que permitirán comprender mejor la composición a nivel físico y lógico del sistema que comprenderá esta plataforma.

* 1. **Lenguajes y Plataformas**

El lenguaje que el equipo ha decidido implementar para el desarrollo de nuestro sistema es el leguaje C# de Microsoft. Consideramos que esta es una opción apropiada debido a la naturaleza del proyecto, permite una importante escalabilidad de nuestra aplicación, permitiendo que el sistema sea mucho más robusto.

En este lenguaje estaremos utilizando lo que es el framework .Net Core en su versión 3.1, para así poder lograr que nuestra aplicación sea multiplataforma, y crear nuestra propia API REST.

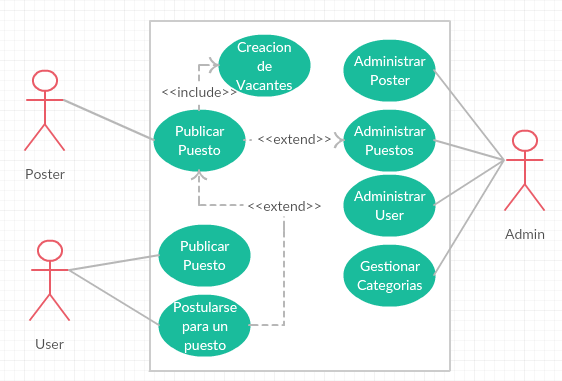
Como gestor de base de datos optamos por solución de SQL Server que también es de Microsoft, esto no ayudara a tener un obtener más de compatibilidad y agilidad en el desarrollo, lo que se traduce en una solución más robusta y eficiente.

Nuestro equipo adoptará la cultura, las prácticas y las herramientas de DevOps, para mejorar el rendimiento y crear nuestro sistema con una mayor calidad en menor tiempo.

Del lado del alto nivel y de la interfaz de usuario se utilizaremos HTML, JS y BOOSTRAP, logrando así una apariencia moderna, interactiva y agradable al usuario.

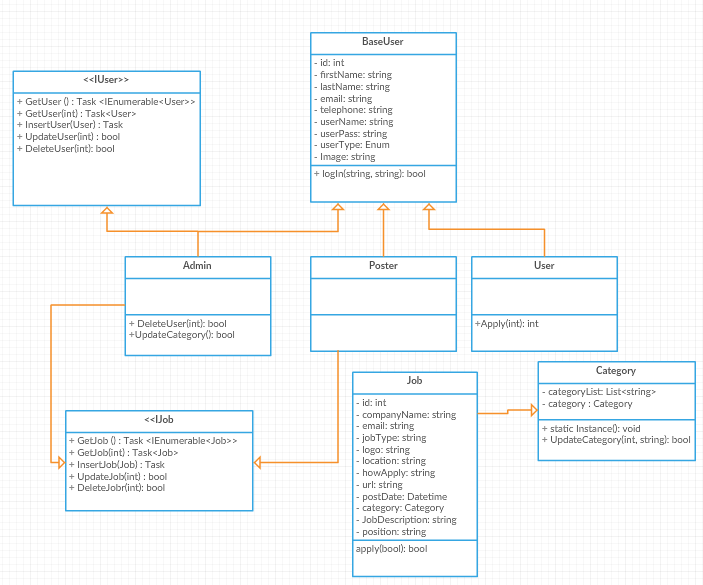
1. ***Vista de Escenarios***

Diagrama de Casos de uso



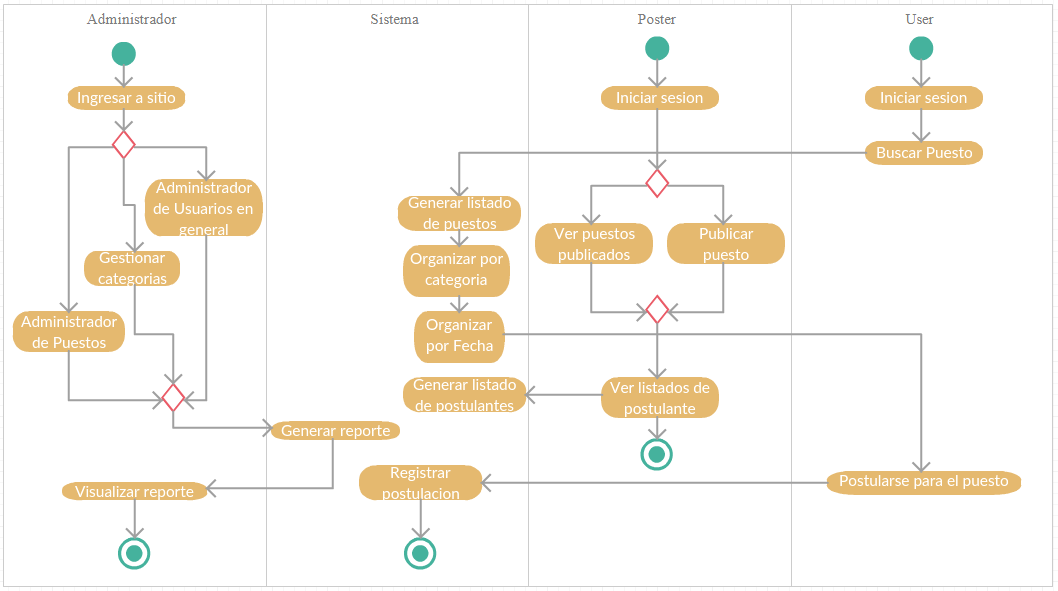
1. **Vista Lógica**

Diagrama de Clases



1. **Vista de Procesos**

Diagrama de Actividades



1. ***Vista de Desarrollo***

Diagrama de Componentes:

