

Programación Web 3

UNLaM - Tecnicatura en Desarrollo Web

Trabajo Práctico de Investigación

Título: Azure

Integrantes	1
Objetivo	1
Situación Actual	2
Desarrollo de la Investigación	2
Conclusiones	2

Integrantes

Espinoza, María Chiara. DNI 42.673.774

Oyaregui, Manuel. DNI 43.082.374

Pereira, Rodrigo Nicolas. DNI 39.464.179

Piedrabuena, Walter Nicolas. DNI 38.266.294

Objetivo

El objetivo principal de la investigación es indagar y mostrar el uso y funcionamiento de múltiples servicios de Azure a través de una aplicación web .NET. El enfoque y alcance de este trabajo de investigación es el uso integrado de Azure Functions, Databases, Azure App Service y el manejo del portal de Azure en general.

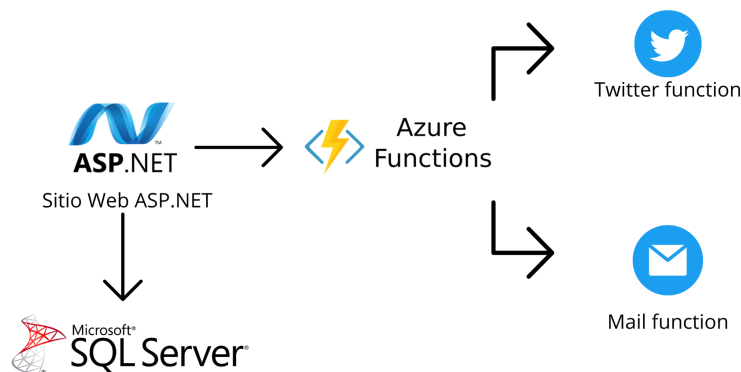
Situación Actual

Hoy en día Azure es uno de los dos servicios en la nube más grandes del mundo, ofreciendo múltiples servicios, aplicaciones y plataformas para facilitar una amplia gama de problemas. Azure facilita el uso de servicios ocupándose de ciertas responsabilidades que, si se realizara on-premise (“en las propias instalaciones”), uno tendría que encargarse de forma manual. Esto permite a las organizaciones y usuarios desarrollar aplicaciones y construir software sin preocuparse de ciertos aspectos técnicos y operativos.

Con el objetivo de demostrar el funcionamiento de los elementos del objeto de investigación, se desarrollará una aplicación ASP.NET integrada con Azure que permita registrar un mensaje de forma anónima y enviárselo a otra persona vía correo electrónico. También se podrá subir a la red social Twitter a una cuenta relacionada con la aplicación. Ambos servicios serán gestionados bajo “Azure Functions”.

En cuanto a las otras tecnologías y servicios a usar tenemos SQL Server para la persistencia de los datos, usando Entity Framework como ORM mediadora.

Arquitectura del proyecto - Tu Secr3to



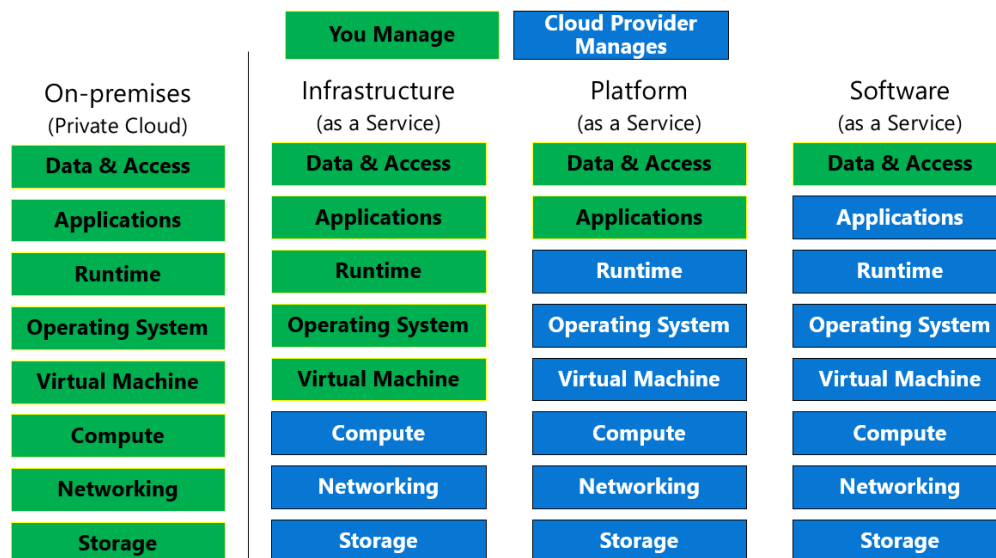
Desarrollo de la Investigación

Azure es una plataforma de servicios en la nube proporcionada por Microsoft que ofrece una amplia gama de soluciones y herramientas para satisfacer las necesidades de las organizaciones en términos de infraestructura, desarrollo y despliegue de aplicaciones. Azure se basa en los modelos de servicio conocidos como IaaS (Infraestructura como Servicio), PaaS (Plataforma como Servicio) y SaaS (Software como Servicio), los cuales desempeñan un papel fundamental en su funcionalidad y flexibilidad.

En primer lugar, Azure ofrece el modelo de **IaaS**, lo que significa que proporciona una infraestructura de TI virtualizada y escalable en la nube. Esto incluye máquinas virtuales, redes, almacenamiento y otros recursos necesarios para ejecutar y administrar aplicaciones. Los usuarios tienen un control completo sobre la configuración y el mantenimiento de la infraestructura subyacente, permitiéndoles implementar sistemas operativos y aplicaciones personalizadas según sus necesidades específicas.

En segundo lugar, Azure ofrece el modelo de **PaaS**, que se centra en proporcionar una plataforma completa para el desarrollo, la implementación y la gestión de aplicaciones. En este caso, los usuarios no necesitan preocuparse por la infraestructura subyacente, ya que Azure se encarga de proporcionar el entorno de ejecución, el middleware y los servicios necesarios para el desarrollo de aplicaciones. Esto permite a los desarrolladores centrarse únicamente en la creación de sus aplicaciones sin preocuparse por la gestión de la infraestructura.

Por último, Azure también ofrece el modelo de **SaaS**, que permite a los usuarios acceder y utilizar aplicaciones de software a través de Internet, sin necesidad de instalar ni mantener el software en sus propios dispositivos. En este caso, Microsoft se encarga de proporcionar y mantener el software, mientras que los usuarios solo necesitan acceder a través de un navegador web u otro cliente de software específico.



Azure es una excelente herramienta con la cual trabajar para desplegar aplicaciones web .NET ya que podemos apoyarnos en esa capacidad de desligarse de responsabilidades, haciendo más fácil el hecho de concentrarse meramente en construir nuestra aplicación lo más efectivamente.

En el caso de este proyecto de investigación, la idea fue crear una aplicación web ASP.NET con una interfaz simple para realizar una carga de datos a través de un formulario, y partiendo de esa base, implementar diversos servicios de Azure. La aplicación web se llama Tu Secr3to y permite escribir mensajes anónimos que serán enviados por mail o por Twitter a la persona que se desee.

Para las funcionalidades de enviar mensajes elegimos usar **Azure Functions**, un servicio para ejecutar código bajo demanda disparado por eventos específicos. Este código está escrito como función independiente en C# y se encuentra 'en la nube' de Azure, de forma serverless (sin servidor) sin tener que preocuparnos por la infraestructura que corre por detrás. Nuestra app .NET llama a la Azure Function y la función se ejecuta. Para las funcionalidades de mandar un mensaje por mail o por Twitter, decidimos crear dos funciones distintas con el propósito de encapsular cada una y separar las responsabilidades: que cada función tenga la mínima cantidad de código necesaria.

Por otro lado, decidimos persistir la información usando **Azure SQL Server** y dentro una **Azure SQL Database**. Azure permite tener servicios de bases de datos (Infraestructura como Servicio) y bases de datos funcionando en sí (Plataforma como Servicio). Ambas son opciones que nos permiten usar bases de datos de forma escalable. Esta es una de las características en las cuales se caracteriza Azure y el mundo de *cloud computing* en sí. Puedo ir escalando y subiendo la cantidad de servidores, bases de datos, o incluso la cantidad de almacenamiento, simplemente aumentándolo desde el portal de Azure.

Con respecto al **portal** (<https://portal.azure.com/>) tiene una interfaz muy intuitiva y fácil de usar, que permite crear recursos simplemente buscandolos. Lo único importante al crear recursos o servicios es que estén contenidos dentro de un Grupo de Recursos, un contenedor lógico para agrupar herramientas dentro de una solución determinada.

Investigando, implementando y sabiendo usar varios servicios de Azure, es posible crear aplicaciones .NET de forma sencilla y aprovechando de todos los beneficios que ofrece la nube.

Conclusiones

Azure resultó siendo una gran herramienta, tal como lo promociona, para desarrollar y delegarle responsabilidades de infraestructura. A pesar de que es fácil verse abrumado, en un principio, por su vastedad de servicios, una vez que separamos la investigación en pequeñas partes, pudimos ver qué herramientas nos eran útiles para demostrar de forma práctica lo que se pidió para este trabajo de investigación.

Los servicios de Azure Function son cómodos para separar responsabilidades y compartimentalizar pequeños fragmentos de código, el portal de Azure es intuitivo y fácil de usar, y con Azure App Service es mucho más fácil visualizar y compartir la aplicación.

Es fácil ver cómo este servicio es potencia en el mundo de las tecnologías en la nube, y logramos ver la utilidad así como también pensamos usarlo en materias a futuro.

Referencias/Bibliografía

Microsoft. (s.f.). **Documentación de Azure**. Recuperado de <https://learn.microsoft.com/es-es/azure/?product=popular>

Microsoft. (s.f.). **Azure Functions documentation**. Recuperado de <https://learn.microsoft.com/en-us/azure/azure-functions/>

GitHub. (s.f.). **Implementación en Azure - Documentación de GitHub**. Recuperado de <https://docs.github.com/es/actions/deployment/deploying-to-your-cloud-provider/deploying-to-azure>

Ibermática 365. (s.f.). **¿IaaS? ¿SaaS? ¿PaaS? Lo que debes saber de Azure**. Recuperado de <https://www.ibermatica365.com/iaas-saas-paas-guia-para-entender-azure/>

Microsoft. (s.f.). **Azure Portal**. Recuperado de <https://portal.azure.com/>

OpenAI (s.f.) **Chat GPT-3**. <https://chat.openai.com/>