

Universidad de Lima
Facultad de Ingeniería y Arquitectura
Carrera de Ingeniería de Sistemas



EXPERIENCIA PROFESIONAL EN EL ÁREA DE DESARROLLO DE SOFTWARE

Trabajo de suficiencia profesional para optar el Título Profesional de Ingeniero de
Sistemas

Moritz Momediano Almonacid

Código 20112047

Asesor

Daniel Enrique Cardenas Salas

Lima – Perú

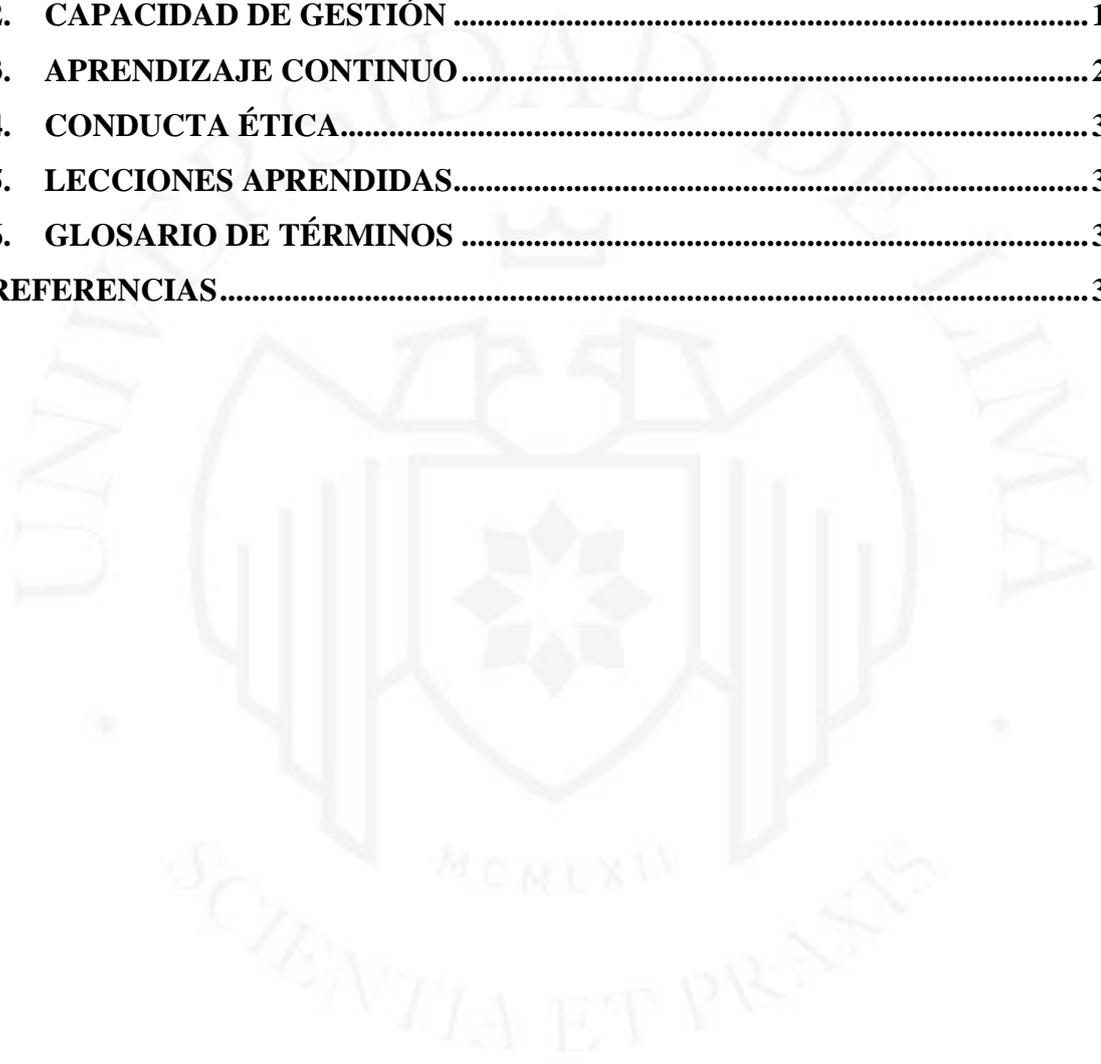
Diciembre de 2022



**PROFESSIONAL EXPERIENCE IN THE
AREA OF SOFTWARE DEVELOPMENT**

TABLA DE CONTENIDO

RESUMEN	vi
ABSTRACT.....	vii
INTRODUCCIÓN	1
1. CAPACIDAD TÉCNICA.....	3
2. CAPACIDAD DE GESTIÓN	17
3. APRENDIZAJE CONTINUO.....	25
4. CONDUCTA ÉTICA.....	31
5. LECCIONES APRENDIDAS.....	34
6. GLOSARIO DE TÉRMINOS	36
REFERENCIAS.....	38



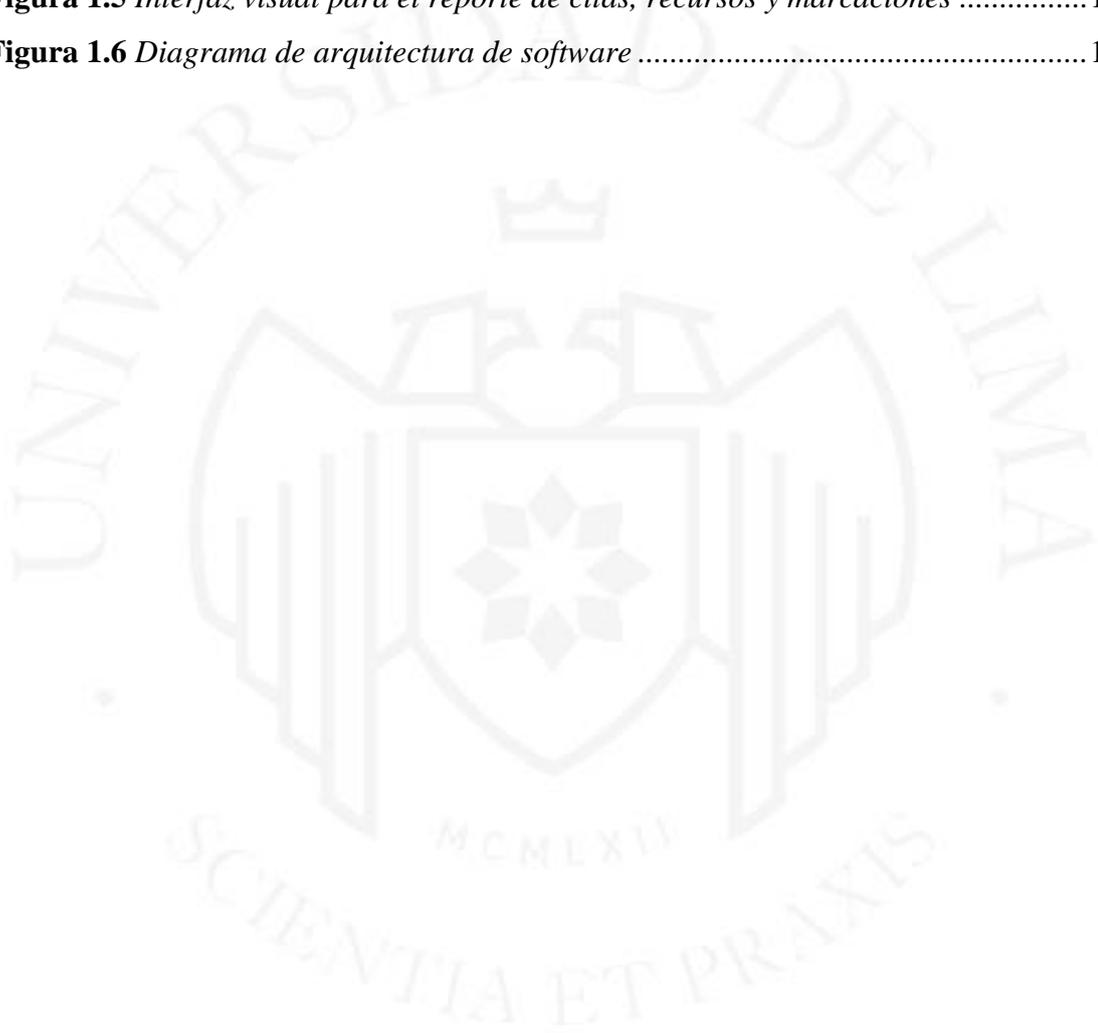
ÍNDICE DE TABLAS

Tabla 1.1 Reporte de tiempos de carga de diferentes escenarios a 100 usuarios	15
Tabla 2.1 Actividades en cada sprint	18
Tabla 2.2 Reporte de horas ahorradas	19
Tabla 2.3 Matriz de riesgo	22



ÍNDICE DE FIGURAS

Figura 1.1 <i>Configuración en switch Cisco</i>	4
Figura 1.2 <i>Gabinete de conexión con switch y patch panel</i>	5
Figura 1.3 <i>Interfaz visual para el registro de citas</i>	9
Figura 1.4 <i>Interfaz visual para el registro de marcaciones</i>	9
Figura 1.5 <i>Interfaz visual para el reporte de citas, recursos y marcaciones</i>	10
Figura 1.6 <i>Diagrama de arquitectura de software</i>	14



RESUMEN

Como profesional egresado de la Universidad de Lima, tuve la oportunidad de afrontar diferentes desafíos en diferentes áreas laborales, empecé en el área de redes y comunicaciones, apoyando en la creación de topologías de red y en la configuración de dispositivos de comunicación, luego en el área de help desk, brindando soporte al sistema operativo de Windows 10 y por último en el área de desarrollo e innovación, donde afronté todos los nuevos retos gracias a mis conocimientos adquiridos en la carrera de ingeniería sistemas en la universidad y el aprendizaje autodidacta sobre las nuevas tecnologías para el desarrollo de software.

Esta última experiencia me permitió tomar una elección sobre mi línea de carrera como desarrollador de software, con la que puedo brindar soluciones tecnológicas e innovadoras, además de participar en el diseño, el desarrollo y la implementación de aplicaciones web o móvil.

Tener la suficiente experiencia técnica en el desarrollo de software, como la automatización de tareas, mejorar los flujos de trabajo, añadir nuevas funcionalidades a los sistemas que utilizaban los diferentes clientes interesados, entre otros, me permitió ser gestor de proyectos como líder técnico. Esta experiencia previa, me permitió participar en la toma de decisiones con distintos equipos, para el logro de sus objetivos, de acuerdo a la necesidad y las metas planteadas.

En este informe detallo mi experiencia laboral, en el cual destaco mi rol de desarrollador, sobre todo mi participación en el análisis y desarrollo de soluciones tecnológicas en los diferentes proyectos en los que participé.

Palabras clave: Desarrollo de software, Diseño de software, Análisis de requerimientos, Gestión de proyectos, Innovación, Scrum.

ABSTRACT

As a professional graduated from the University of Lima, I had the opportunity to face different challenges in different work areas, I started in the area of networks and communications, supporting the creation of network topologies and the configuration of communication devices, then in the area of help desk, providing support to the Windows 10 operating system and finally in the area of development and innovation, where I faced all the new challenges thanks to my knowledge acquired in the systems engineering career at the university and self-taught learning about new technologies for software development.

This last experience allowed me to make a choice about my career as a software developer, with which I can provide technological and innovative solutions, as well as participate in the design, development and implementation of a web or mobile application.

Having enough technical experience in software development, such as automating tasks, improving workflows, adding new functionalities to the systems used by the different interested clients, among others, allowed me to be a project manager as a technical leader. This previous experience allowed me to participate in decision-making with different teams, to achieve their objectives, according to the need and the goals set.

In this report I detail my work experience, in which I highlight my role as a developer, especially my participation in the analysis and development of technological solutions in the different projects in which I participated.

Keywords: Software development, Software design, Requirements analysis, Project management, Innovation, Scrum.

INTRODUCCIÓN

El egresado de la carrera de Ingeniería de Sistemas de la Universidad de Lima es capaz de afrontar diversos desafíos, en las diferentes áreas laborales, diseñando e implementando sistemas que permiten el logro de objetivos, utilizando soluciones innovadoras basadas en tecnologías de información. Además, el ingeniero de sistemas tiene la habilidad para identificar y entender las necesidades de una organización, para tomar decisiones que permitan cubrir el alcance requerido para satisfacer dichas necesidades.

Mi experiencia laboral demuestra que, como ingeniero de sistemas de la Universidad de Lima, tengo los conocimientos necesarios para afrontar los retos que se me presentaron en las diferentes especialidades de la carrera.

Mi experiencia comenzó en el 2017 con el rol de analista en redes y comunicaciones, brindando soporte en cableado estructurado, soporte en dispositivos de comunicación, realizando reportes de tráfico de red, inventarios de equipos y accesorios de administración de red.

Luego, en 2018, fui asistente de help desk, brindando soporte en el sistema operativo Windows para la solución de errores en el proceso de actualización del sistema operativo en varias empresas en el sector financiero.

Por último, en el mismo año, inicié mi línea de carrera como desarrollador de software, donde aprendí a trabajar en equipo para lograr la creación de una aplicación, luego aprendí a crear pruebas unitarias para mantener la calidad de la aplicación, y finalmente a realizar mantenimiento para poder corregir errores o realizar mejoras.

Los diferentes retos propuestos durante toda mi experiencia laboral me han permitido afirmar mis conocimientos adquiridos en la Universidad de Lima. Además, al inicio y final de cada reto planteado me ha permitido evaluar mis competencias, y gracias al aprendizaje continuo he estado preparado para afrontar nuevos retos.

En este informe presento algunos de los proyectos en los que participé, detallando la necesidad o el objetivo que se deseó lograr por las partes interesadas, y la solución que se planteó de acuerdo con el equipo técnico con el que colaboré. Además, indico las

habilidades técnicas y blandas adquiridas para cumplir con las expectativas deseadas, durante el desarrollo del proyecto.



1. CAPACIDAD TÉCNICA

Mi experiencia en la carrera de Ingeniería de Sistemas inició en el 2015 en el área de redes y comunicaciones en la Universidad de Lima. En ese año se determinó que varios de los dispositivos de comunicación que se tenían hasta la fecha estaban desactualizados o deteriorados. Por ejemplo, se usaban repetidores (hubs) deteriorados para establecer la conexión entre los dispositivos, o el cableado estructurado de red y el cableado eléctrico compartían el mismo espacio, por lo que ocasionaba pérdida de paquetes de conexión en determinados momentos. Por ello, participé en el proyecto de renovación de infraestructura de Tecnología de Información (TI), que se refiere al conjunto de tecnologías, sistemas y herramientas utilizados para gestionar, almacenar, procesar y transmitir información. El objetivo era segmentar la infraestructura de TI de los pabellones de Administración, Psicología y Comunicaciones, reemplazar los dispositivos de comunicaciones (switches, access points y/o hubs) desactualizados, y configurar nuevos dispositivos de comunicación que serían asignados a los nuevos pabellones.

Durante el proyecto realicé diferentes funciones:

- a) Realicé configuraciones en los switches de marca Cisco para crear una Red de Área Local Virtual (VLAN), la cual es una tecnología que permite dividir una red física en múltiples segmentos lógicos.
- b) Asigné a cada nuevo dispositivo una dirección única para su identificación, utilizando el Protocolo de Internet (IP) y una máscara de red. Además, los agregué a una red VLAN, de acuerdo al piso y pabellón al que iban a ser asignados. (Véase Figura 1.1)

Figura 1.1

Configuración en switch Cisco

```
Switch(config)# interface vlan 1
Switch(config-if)# ip address 192.168.1.100 255.255.255.0
Switch(config-if)# no shutdown
12:35.07: %LINK-3-UPDOWN: Interface Vlan1, changed state to up
12:35.08: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan 1, changed
state to up
Switch(config-if)# exit
Switch(config)# ip default-gateway 192.168.1.1
```

Nota. La figura es una referencia a los comandos que se utilizó para configurar los switches.

- c) Realicé el cableado desde el área académica donde se encontraban las computadoras, access points y/o impresoras hasta los gabinetes de comunicación donde se encontraban los switches.
- d) Elaboré patch cords, usando conectores RJ-45 hembras en un extremo de los cables UTP CAT 5 para la conexión de los switches, los cuales se encontraban en los gabinetes de conexión. En el otro extremo, utilicé conectores RJ-45 hembras junto a un faceplate para realizar la conexión hacia los diferentes ambientes. Por último, se validó la distancia y el estándar de conexión que se requería con un medidor de señal digital.
- e) En los gabinetes de conexión previamente instalados en cada pabellón, instalé los switches y patch panels que fueron asignados a dicho pabellón. En los patch panels se colocaron los cables con los RJ-45 hembras que provenían de los diferentes ambientes académicos, y luego se realizaban las conexiones desde el mismo patch panel hasta los switches ubicados en el mismo gabinete de conexión utilizando cables Ethernet de 1 metro y medio. (Véase Figura 1.2)

Figura 1.2

Gabinete de conexión con switch y patch panel



Nota. La figura es una referencia a la manera de cómo se ordenan y conectan los switches a los patch panel ubicados en un gabinete de conexión.

Luego de realizar toda la implementación, se llevaron a cabo diferentes validaciones, tales como:

- Verificar que el tráfico de red sea estable, enviando paquetes de mensajes usando el protocolo de control de mensajes de internet (ICMP) a las diferentes direcciones IP y comprobando que el tiempo de respuesta sea menor a 100 milisegundos.
- Comprobar que la señal de WiFi sea mayor a -90 dbm, para asegurar una excelente cobertura de los access points en cada área instalada.

Por último, se realizó un control de inventario de todos los equipos que fueron renovados y que se encontraban en buen estado para poder ser reutilizados cuando sea necesario.

En conclusión, la instalación de nuevos dispositivos de red mejoró la comunicación entre los diferentes pabellones, tanto a través de conexiones por cable como inalámbricas. Además, se logró solucionar los problemas de conexión de manera más eficiente gracias a la actualización de la información sobre la ubicación de las

conexiones de los diferentes dispositivos de comunicación y al orden adecuado de las conexiones en los gabinetes de comunicaciones.

En el 2018, tuve mi segundo reto en una consultora de TI que, como socio de Microsoft, brindaba servicios de soporte relacionados con este. Una empresa del sector financiero solicitó sus servicios para actualizar el sistema operativo de Windows XP a Windows 7 en todos los equipos de sus sucursales ubicadas en diferentes regiones de Perú.

En este proyecto apliqué los conocimientos que adquirí en el área de redes y comunicaciones en la Universidad de Lima. Configuré un puerto de conexión en un switch para poder conectar una laptop que desempeñaba la función de servidor, encargada de ejecutar, monitorear y controlar la actualización de los sistemas operativos de varias computadoras al mismo tiempo. Para llevar a cabo estas tareas, se utilizó la herramienta desarrollada por Microsoft llamada System Center – Orchestrator, la cual ayuda a:

IT pros and/or infrastructure developers can create repeatable automation of repetitive or error prone IT processes in the form of Orchestrator runbooks. Orchestrator runbooks are conceptually similar to scripts in that they perform some set of operations in a repeatable manner. [Los profesionales de TI y/o los desarrolladores de infraestructura pueden crear automatizaciones repetibles de procesos de TI que suelen ser repetitivos o propensos a errores, en forma de runbooks de Orchestrator. Los runbooks de Orchestrator son conceptualmente similares a los scripts, ya que realizan un conjunto de operaciones de manera repetible.] (Ziembicki et al., 2013, p. 5).

Luego de mi experiencia en el área de redes y comunicaciones, a finales del 2018 inició mi experiencia como desarrollador en una empresa que brindaba servicios de TI. Allí aprendí todo lo que sé hasta la fecha, como los diferentes lenguajes de programación, frameworks y/o librerías para el desarrollo de front end y back end, conocimiento en base de datos, infraestructura Cloud, gestión de proyectos con la metodología Scrum, entre otros.

El primer proyecto en el que participé como desarrollador fue para un operador logístico, cuyo objetivo es optimizar los procesos involucrados desde la adquisición de un producto hasta la distribución al cliente final.

La información relacionada con el ingreso y salida del personal autorizado en la empresa se registraba en documentos manuales por parte del personal de seguridad. Estos documentos sólo contenían los datos personales de la persona autorizada a ingresar, así como la fecha y hora de ingreso y de salida. Sin embargo, no se llevaba un control preciso de las áreas a las que la persona ingresaba dentro del establecimiento.

Luego, estos documentos escritos eran registrados en documentos digitales como hojas de Excel. Sin embargo, esto en ocasiones generaba una inconsistencia en los datos, ya que no siempre se registraban todos los documentos o los datos ingresados eran incorrectos.

El objetivo del proyecto era optimizar el proceso de control de accesos, y como solución se decidió desarrollar una aplicación que permitiera monitorear tanto los accesos de los transportes de carga de los productos como los accesos de los usuarios a las diferentes áreas dentro del establecimiento.

En la primera etapa nos enfocamos en entender la necesidad del cliente y elaborar un prototipo de la solución, para ellos se empleó la metodología de Design Thinking, que según Tim Brown, actual CEO de IDEO, “es una disciplina que usa la sensibilidad y métodos de los diseñadores para hacer coincidir las necesidades de las personas con lo que es tecnológicamente factible y con lo que una estrategia viable de negocios puede convertir en valor para el cliente, así como en una gran oportunidad para el mercado” (Brown, 2008, p. 4).

En esta etapa se definieron las necesidades del cliente, y en conjunto con una experta en Experiencia de Usuario e Interfaces de Usuario (UX/UI) se elaboró un prototipo, mediante el cual el cliente podía interactuar y evaluar si cumplía con el objetivo del proyecto.

Para la segunda etapa, junto al equipo técnico del proyecto se definieron las tecnologías que se emplearían y la infraestructura en la que se desplegaría la aplicación:

- Como parte de front end, se utilizó el framework Angular 12.1.2, para cumplir con la necesidad de que la interacción del usuario con la aplicación sea fluida. El framework permite crear una aplicación de página única, lo que significa que los recursos se cargan una sola vez o de manera dinámica según sea necesario, sin necesidad de recargar la página.

- Como parte de back end se utilizó el framework Django 2.0.7, ya que posee el modelo de programación Object-Relational Mapping (ORM), que permite mapear la estructura de una base de datos relacional en una estructura lógica de entidades y administrar las diferentes entidades, creadas desde su interfaz visual como administrador, además, protege las peticiones hacia la base de datos como la inyección SQL o el Cross-Site Request Forgery (CSRF).
- Por último, como base de datos se implementó PostgreSQL v.10, ya que tiene un mejor soporte con la concurrencia de conexiones, ya que el registro de datos desde la aplicación se realizaría desde diferentes áreas y con múltiples usuarios.

En las Figuras 1.3, 1.4 y 1.5, se puede visualizar las interfaces de usuario.



Figura 1.3

Interfaz visual para el registro de citas

N° Cita	Fecha de ingreso	Estado	Visitado	Organización	Visitante	Sede	Zona	Tipo
16543	10-09-2018	pendiente	Jose Tello	TOR UX SAC	Pedro Ramirez(65254247)	Sede principal	Almacen	C
16542	10-09-2018	pendiente	Jose Tello	Global Data	Mariana Lopez(45014674)	Sede principal	Almacen	C
16541	10-09-2018	pendiente	Carolina Ortega	Neo Geo	Carlos Mallqui(25417481)	Sede principal	Oficina central	B
16540	10-09-2018	pendiente	Sergio Bellido	TOR UX SAC	Moises Aguirre(74395626)	Sede principal	Almacen	C
16539	10-09-2018	pendiente	Jose Tello	Global Data	Sofia Quispe(12582435)	Sede principal	Almacen	C
16538	10-09-2018	pendiente	Carolina Ortega	Neo Geo	Ximena Sanchez(45096277)	Sede principal	Oficina central	B
16537	10-09-2018	cancelado	Jose Tello	TOR UX SAC	Fernando Tipe(16035377)	Sede principal	Almacen	C
16536	10-09-2018	pendiente	Francisco Ludeña	Neo Geo	Rodrigo Jauregui(99667335)	Sede principal	Area de recursos humanos	A
16535	10-09-2018	cancelado	Francisco Ludeña	Bit SAC	Andrea Rivera(92205027)	Sede principal	Area de recursos humanos	A
16534	10-09-2018	pendiente	Carolina Ortega	Bit SAC	Harold Aquino(25974278)	Sede principal	Oficina central	B

Nota. En la interfaz se visualiza el registro de citas agendadas y el detalle de las citas.

Figura 1.4

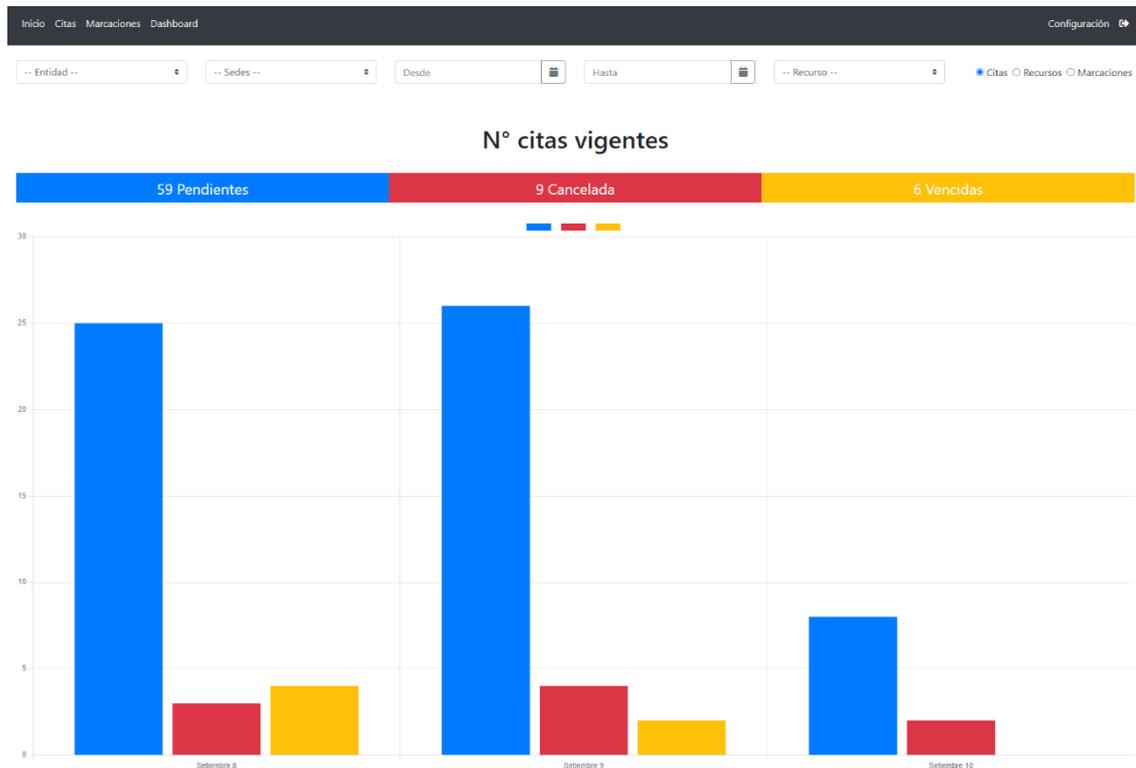
Interfaz visual para el registro de marcaciones

Fecha de Cita	Entidad	Recurso	Ubicación	Tipo	Fecha de marcación
10-09-2018	TOR UX SAC	Pedro Ramirez	Almacen	C	10-09-2018 17:38
10-09-2018	Global Data	Mariana Lopez	Almacen	C	10-09-2018 17:12
10-09-2018	Neo Geo	Carlos Mallqui	Oficina central	B	10-09-2018 16:48
10-09-2018	TOR UX SAC	Moises Aguirre	Almacen	C	10-09-2018 13:23
10-09-2018	Global Data	Sofia Quispe	Almacen	C	10-09-2018 13:01
10-09-2018	Neo Geo	Ximena Sanchez	Oficina central	B	10-09-2018 12:33
10-09-2018	TOR UX SAC	Fernando Tipe	Almacen	C	10-09-2018 12:08
10-09-2018	Neo Geo	Rodrigo Jauregui	Area de recursos humanos	A	10-09-2018 10:38
10-09-2018	Bit SAC	Andrea Rivera	Area de recursos humanos	A	10-09-2018 10:14
10-09-2018	Bit SAC	Harold Aquino	Oficina central	B	10-09-2018 09:52

Nota. En la interfaz se visualiza el registro de fecha y hora de cada usuario como control de acceso.

Figura 1.5

Interfaz visual para el reporte de citas, recursos y marcaciones



Nota. En la interfaz se visualiza el reporte de citas agendadas por día y estado de la cita.

En la última etapa del proyecto, se realizó una capacitación a los usuarios finales, donde se les explicó el uso de la aplicación y las características que tenía por cada tipo de rol.

En conclusión, el desarrollo de la aplicación permitió un control más detallado sobre el acceso del personal autorizado dentro del establecimiento, como el registro de la hora y fecha en las áreas ingresadas por el personal externo. Además, se contó con un reporte sobre la cantidad de personas que registraban o cancelaban una cita, el historial de ubicaciones que un usuario visitaba, entre otros aspectos.

En el 2019, formé parte del equipo de desarrollo para una empresa del sector metalúrgico. Participé en la tercera fase del proyecto con el objetivo de crear automatizaciones para la gestión de ideas e iniciativas en la plataforma Jira Server versión 7, que fue solicitado por su área de Resource and Development.

La plataforma Jira permite administrar las actividades de un proyecto y el seguimiento del estado de las tareas correspondientes. También permite agregar plugins,

que mediante su desarrollo permite añadir nuevas funcionalidades a Jira. El desarrollo de estos plugins se lleva a cabo utilizando el framework creado por Atlassian basado en el lenguaje Java 8. Este framework implementa la herramienta Maven para la gestión de librerías y la construcción de los plugins.

El objetivo principal de la primera y segunda fase fue automatizar los procesos para la evaluación de nuevos proyectos considerados como iniciativas. Una iniciativa es una propuesta de innovación para mejorar algún proceso interno de la organización o crear un nuevo producto.

En estas fases iniciales se agregaron diferentes automatizaciones en el flujo de trabajo para una iniciativa, desde la creación de las tareas consideradas como iniciativas hasta su aceptación. Algunas de estas automatizaciones son:

- Validación de la propuesta de la iniciativa.
- Asignación de usuarios en las diferentes etapas del proceso.
- Actualización de los datos relevantes de la iniciativa, de acuerdo a la información requerida en cada etapa del proceso.
- Generación de costos.
- Evaluación del retorno de inversión, entre otros.

Para la tercera fase, fue necesario generar nuevos proyectos y flujos de trabajo para cada iniciativa que fue aceptada y ahora se consideraban como ideas de innovación.

Durante esta fase, brindé el apoyo evaluando los procesos y las posibles mejoras a través del mapeo de procesos AS IS/TO BE. Además, participé en la configuración de los esquemas del proyecto dentro de Jira.

Al realizar la configuración de los esquemas en los proyectos, se desarrolló plugins que añadían nuevas funcionalidades a las transiciones, las cuales representan “The path that takes an issue from one status to the next” [el camino que lleva una tarea de un estado al siguiente] (Li, 2016, p. 189)

En cada transición se implementaron:

- Conditions: “Criteria must be met before the transition is available (visible) for users to execute. It is usually used to control permissions around how users can execute the transition.” [Se deben cumplir los criterios antes de que la

transición esté disponible (visible) para que los usuarios la ejecuten. Por lo general, se usa para controlar los permisos sobre cómo los usuarios pueden ejecutar la transición.] (Li, 2016, p. 189)

- Validators: “These are the verifications that must pass before the transition can be executed. They are usually used together with Transition Screens.” [Estas son las verificaciones que deben pasar antes de que la transición pueda ser ejecutada. Por lo general, se usan junto con las pantallas de transición.] (Li, 2016, p. 189)
- Post Functions: “These are additional functions to be performed as part of the transition process.” [Estas son funciones adicionales que se realizarán como parte del proceso de transición.] (Li, 2016, p. 189)

Al finalizar el proyecto, se realizaron pruebas que abarcan desde la creación de una iniciativa hasta su consolidación como un proyecto de innovación. Las automatizaciones en cada estado del proceso permitieron actualizar y generar la información necesaria para que los usuarios puedan evaluar cada iniciativa.

Este proyecto me ayudó a mejorar mis conocimientos sobre la plataforma Jira, como por ejemplo, configurar proyectos, administrar permisos y roles, mejorar procesos de trabajo y crear automatizaciones.

A inicios del año 2020, participé en el desarrollo de una aplicación para una empresa del sector de educación, donde la empresa requería mejorar su plataforma de procesos de matrícula.

La aplicación web de matrícula online con la que contaban era poco intuitiva para el usuario, por lo que recurrían a las diferentes sedes para realizar el proceso de matrícula de manera presencial. Para matricular a un usuario de manera online, se requería seguir demasiados pasos innecesarios e ingresar datos de manera repetitiva que ya habían sido solicitados en pasos anteriores o en matrículas pasadas.

Para ello, en la primera etapa se evaluaron las siguientes acciones:

- Rediseñar la interfaz visual de la página web de matrícula online, ya que resultaba poco comprensible para el usuario final.
- Mantener los procesos actuales debido a que estaban integrados con diferentes plataformas, como el registro de pagos, el registro de matrículas, entre otras.

Debido a los grandes cambios que se requerían y a la necesidad de rediseñar la arquitectura del software, se tomó la decisión de crear una nueva aplicación, manteniendo el modelo de datos actual en la base de datos y la integración con las diferentes plataformas de registro de datos. (Véase Figura 1.6)

En la segunda etapa se diseñó la arquitectura del software:

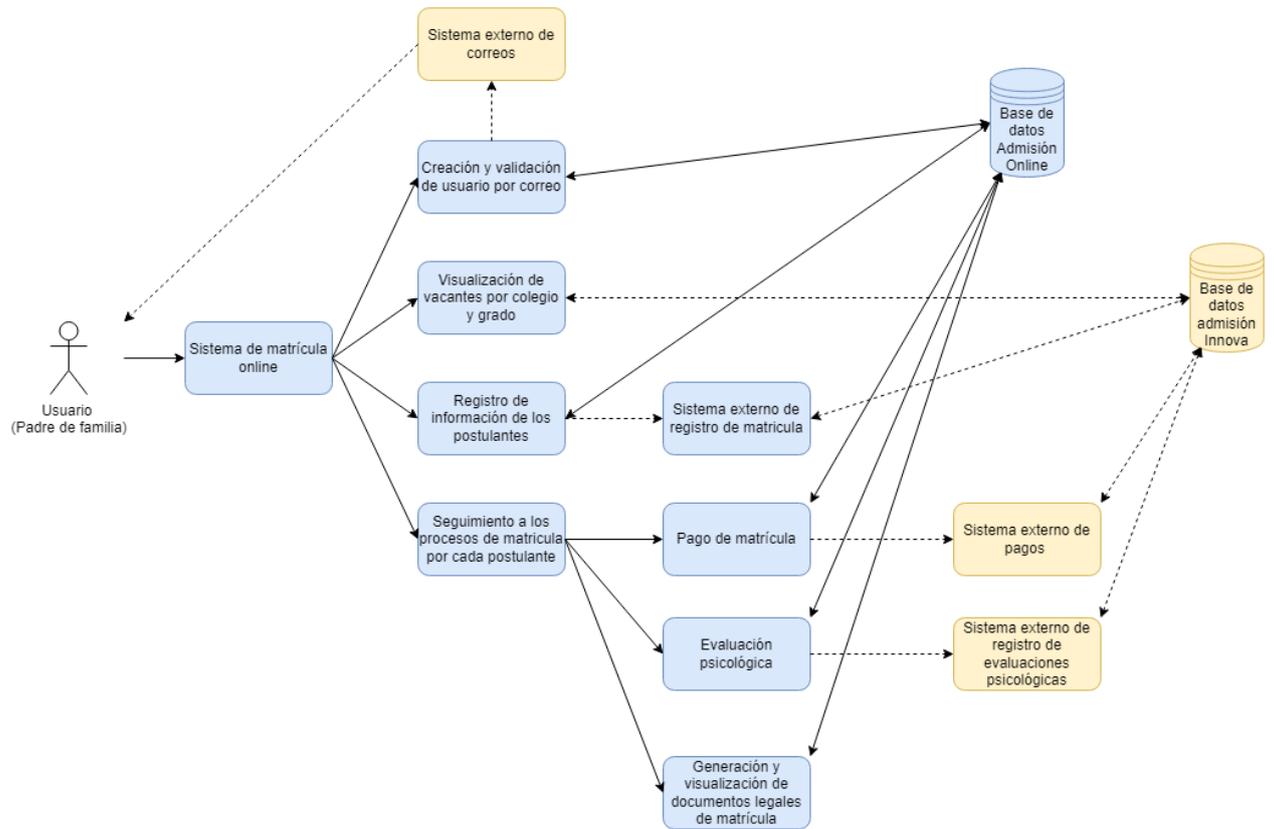
- Se creó la arquitectura con el framework ASP.NET 2017, con el lenguaje C#.
- Se implementó el patrón de diseño Model-View-Controller (MVC), el cual se caracteriza por enfatizar la separación entre la lógica de negocios y la visualización.
- Se utilizó el framework AngularJS para poder crear y mantener la aplicación como una sola página web, sin tener la necesidad de refrescar toda la información cuando cambia una vista.
- Se utilizó la característica Object Relational Mapping (ORM) del framework C# Entity para SQL server que permitía la administración de la base de datos desde la aplicación.

Para la tercera etapa de desarrollo de la aplicación, se dividió en 4 fases:

- En la primera fase se realizó la implementación del registro de la información del padre y la búsqueda de vacantes para los hijos en las diferentes sedes.
- En la segunda fase se realizó la implementación del registro de la información de los hijos y la matrícula en el grado y sección correspondiente.
- En la tercera fase se implementó una interfaz que permitía a los padres dar seguimiento al proceso de matrícula por cada hijo matriculado. Además, se realizó la implementación de pagos con el Interfaz de Programación de Aplicaciones (API) de Visa.
- Por último, se realizó una interfaz que permitiera agendar o realizar evaluaciones psicológicas.

Figura 1.6

Diagrama de arquitectura de software



Nota. Diagrama de arquitectura de software del sistema de matrícula online.

Para la última etapa se realizaron pruebas de carga con 100 usuarios en diferentes escenarios (véase Tabla 1.1), con la herramienta JMeter, que permitió realizar pruebas de estrés y validar el rendimiento estático y dinámico de los recursos.

Tabla 1.1*Reporte de tiempos de carga de diferentes escenarios a 100 usuarios*

Escenario	Tiempo de respuesta promedio (ms)	Tiempo de respuesta mínimo(ms)	Tiempo de respuesta máximo(ms)
Búsqueda de vacante sin inicio de sesión, registro de padre de familia, y registro del estudiante hasta estar matriculado.	1051.83	3	7191
Búsqueda de tres vacantes sin inicio de sesión, registro de padre de familia, y registro de los tres estudiantes hasta estar matriculados.	1181.36	3	8918
Inicio de sesión, búsqueda de una vacante, y registro del estudiante hasta estar matriculado.	895.81	5	5016

Nota. Reporte con la herramienta JMeter.

En conclusión, la aplicación web permitió a los usuarios poder realizar la matrícula online de manera más intuitiva, seguir los pasos de manera secuencial y evitar la repetición de información. Además, la aplicación ofrece un sistema de registro de las etapas del proceso de matrícula, lo que permite al usuario final dar seguimiento a cada etapa, desde el registro de los alumnos hasta la confirmación de la matrícula.

En el 2021, formé parte del equipo del proyecto de Intranet para una empresa del sector educativo. Debido a la pandemia del COVID-19, se necesitaba una plataforma para realizar diferentes procesos que pudieran evitarse de manera presencial.

El objetivo del proyecto era:

- Brindar a los colaboradores información relevante como eventos, noticias y/o anuncios a través de un panel principal.
- Administrar los perfiles de cada colaborador.
- Asignar roles de trabajo con los permisos necesarios dentro de la plataforma.
- Realizar diferentes procesos, como la actualización y validación de información personal, asignación de equipos de trabajo, asignación de roles de trabajo, entre otros.

Durante la etapa de desarrollo de este proyecto, apliqué mis conocimientos sobre la librería React, donde:

- Brindé recomendaciones sobre la creación de la estructura de los componentes, ya sea como clases o funciones según la necesidad del componente.
- Refactoricé las componentes para que cumplieran el objetivo de ser reutilizables.
- Generé casos de prueba de los componentes creados, con el fin de validar tanto su funcionalidad como su renderización.

Asimismo, brindé apoyo en el desarrollo de APIs con el framework ASP.NET y Entity Framework para la creación, lectura y actualización de información en la base de datos.

En conclusión, se agregaron nuevas funcionalidades para que los colaboradores eviten realizar procesos presenciales debido a la pandemia. Además, se realizaron mejoras en el código y se agregaron pruebas unitarias para mejorar la calidad del proyecto desarrollado.

2. CAPACIDAD DE GESTIÓN

Mi primera experiencia gestionando un proyecto fue a mediados del 2020 en un proyecto de mejoras para el proceso de ventas de una empresa que brindaba servicios de consultoría.

Su proceso de ventas, administrado a través de la plataforma Jira, contaba con procesos largos y repetitivos que no generaban mucho valor en comparación con el tiempo que tomaba realizarlos, ya que se trataba de un proceso manual.

Como solución, se decidió desarrollar plugins para la plataforma Jira que permitieran automatizar las diferentes tareas.

Para el proyecto, se empleó la metodología Scrum. Como primer paso, se definieron los siguientes actores:

- Al líder de equipo del área de ventas como product owner.
- A la líder del proceso de ventas como stakeholder.
- A mí como Scrum Master y parte del equipo de desarrollo.

Como segundo paso, se definió como objetivo ahorrar al menos el 40% de las horas mensuales y, junto con una asistente de ventas que conocía todo el proceso de ventas, se definió el backlog con las siguientes actividades:

- Generar una página en la plataforma Confluence con la información del cliente que deseaba una propuesta de venta.
- Generar documentos en la plataforma Confluence utilizando una plantilla con la información sobre la oportunidad de venta.
- Generar documentos en la plataforma Confluence utilizando una plantilla para documentar la propuesta técnica y la propuesta económica que se iba a ofrecer al cliente.
- Generar documentos en Google Sheets para crear la estructura de costos de la oportunidad de ventas.
- Actualizar la oportunidad de venta creada como tarea en Jira, con la información de la estructura de costos generada en Google Sheets.

- Actualizar los enlaces, los cuales hacen referencia a todos los documentos generados en Confluence y Google Sheets, en la oportunidad de venta generada como tarea en Jira.

En la etapa de planificación, se utilizó la plataforma Jira para gestionar las actividades mencionadas en el backlog, realizar el seguimiento y conocer el estado actual del proyecto.

En esta etapa, con la participación del stakeholder y product owner, se definieron tres sprints de una semana cada una. (Véase Tabla 2.1)

Tabla 2.1

Actividades en cada sprint

Sprint	Actividades
1	<ul style="list-style-type: none"> • Generar una página en la plataforma Confluence con la información del cliente que deseaba una propuesta de venta. • Generar documentos en la plataforma Confluence utilizando una plantilla con la información sobre la oportunidad de venta.
2	<ul style="list-style-type: none"> • Generar documentos en la plataforma Confluence utilizando una plantilla para la documentación de la propuesta técnica y la propuesta económica que se le iba a ofrecer al cliente. • Generar documentos en Google Sheets para crear la estructura de costos de la oportunidad de ventas.
3	<ul style="list-style-type: none"> • Actualizar la oportunidad de venta creada como tarea en Jira, con la información de la estructura de costos generada en Google Sheets. • Actualizar los enlaces, los cuales hacen referencia a todos los documentos generados en Confluence y Google Sheets, en la oportunidad de venta generada como tarea en Jira.

Nota. Definición de actividades por sprint.

En la etapa de desarrollo, se implementó un plugin que permitía agregar una postfunction en la plataforma de Jira. Esta postfunction se ejecutaba cuando una tarea de Jira realizaba una transición, lo cual cambiaba el estado actual de la tarea y automatizaba las actividades mencionadas en cada sprint.

Al finalizar cada sprint, se realizaba una presentación al product owner y al stakeholder sobre las automatizaciones realizadas en dicho sprint y en conjunto se evaluaban los resultados obtenidos.

Como líder del proyecto, durante el desarrollo del primer sprint, identifiqué que no se contaba con un ambiente de desarrollo, lo cual fue un impedimento para el desarrollo del proyecto, ya que no era una buena práctica realizar las pruebas y demostraciones en su plataforma de Jira y Confluence, que estaban desplegadas en su ambiente de producción. Como solución, se realizó una copia del ambiente de producción tanto de la plataforma Jira como de Confluence en un nuevo ambiente de pruebas para simular los flujos de trabajo del ambiente de producción.

Como última etapa del proyecto, se realizó un reporte en el que se concluyó que, en promedio, se realizaban 680 propuestas al año. Para cada propuesta, se invertían 49 minutos en realizar las diferentes actividades mencionadas previamente. Después de agregar las automatizaciones, el tiempo de actividades se redujo a 11 minutos. De esta manera se realizó una estimación para el año siguiente, obteniendo un ahorro de 413 horas anuales en su proceso de venta. (Véase Tabla 2.2)

Tabla 2.2

Reporte de horas ahorradas

	2019	Hasta agosto del 2020	Desde agosto del 2020	2021+
Cantidad de propuestas	680	395	285	680
Minutos invertidos por propuesta	49	49	11	11
Horas totales invertidas por propuesta	555	322	52	142

Nota. Descripción de las horas ahorradas por cantidad de propuestas al año.

Como primera experiencia liderando un proyecto, aprendí que lo importante del éxito de un proyecto es entender la necesidad del cliente, conocer la situación actual en la que se encuentra y saber cuál es el objetivo al que quiere llegar.

Otra de mis experiencias como gestor de proyecto y líder técnico inició en el año 2021, para un distribuidor automovilístico interesado en el desarrollo de servicios y soluciones digitales. Al invertir en diferentes servicios de tecnología, requerían:

- Evaluar la satisfacción de los empleados respecto a los proyectos a los que eran asignados.
- Evaluar el ambiente laboral de sus empleados.
- Evaluar la satisfacción de los empleados dentro de sus equipos de trabajo.

La empresa ya empleaba la metodología Scrum, por tal motivo como primer paso se definieron los siguientes actores dentro del proyecto:

- El Head of Digital Delivery Centre como product owner.
- El Project Manager of Digital Delivery Centre como stakeholder.
- El team leader del área de servicio, al que pertenecía, como Scrum Master.
- Yo, como líder del equipo de desarrollo.

Como parte de la etapa de planificación, se generó el backlog con las siguientes actividades, considerando que administraba todos sus proyectos en la plataforma Jira:

- Enviar encuestas de satisfacción cada semana, a todos los empleados que participaban más de 20 horas a la semana, sobre el proyecto que era asignado.
- Enviar encuestas de satisfacción a todos los empleados cuando el proyecto finalizaba.
- Generar reportes de satisfacción acorde al rol asignado, al grupo al que pertenecía, al tipo de proyecto y la cantidad de horas invertidas, utilizando el plugin eazyBI Reports and Charts for Jira (Atlassian, 2022b)
- Generar encuestas usando el servicio de SurveyMonkey (Momentive, 2022), que permite generar plantillas por tipo de encuesta, responder las encuestas a través de su servicio y almacenar las respuestas de la encuesta.
- Integrar el plugin Tempo Timesheets for Jira (Tempo, 2022) para conocer los roles y grupos asignados a los proyectos, además del registro de horas de cada trabajador.

Para iniciar el proyecto, se establecieron las siguientes pautas:

- Se dividió el proyecto en 12 sprints para la entrega de la aplicación.
- Cada sprint tendría una duración de 1 semana.
- Se programarían reuniones diarias con el stakeholder para resolver dudas o impedimentos durante la etapa de desarrollo.
- Se realizarían demostraciones como parte de la revisión de cada sprint con el product owner.

Para la gestión del proyecto se utilizó la plataforma Jira:

- Se creó un proyecto dentro de la plataforma con la información del cliente y nombre del proyecto.
- Se configuró los permisos por cada rol asignado a cada participante del proyecto, para la gestión de tareas.
- Se crearon las tareas definidas en el backlog, por cada rol asignado para darles seguimiento en cada sprint.

Para la documentación del proyecto se utilizó Confluence:

- Se creó un espacio donde se documentó la información del cliente.
- Se crearon páginas con la propuesta aceptada del cliente y la solución propuesta por el equipo técnico.
- Se crearon páginas para documentar las reuniones con el cliente y las actas de aceptación al finalizar el proyecto.

Para la gestión de la aplicación se utilizó Bitbucket como repositorio de código fuente que:

- Permite el control de versiones de la aplicación.
- Permite la revisión de cada nuevo requerimiento agregado o mejora realizada en la aplicación.

Para la integración continua se utilizó Bamboo que:

- Permite la compilación de la aplicación.
- Permite el despliegue continuo de la aplicación.

Durante la etapa de desarrollo se identificaron dos riesgos que fueron clasificados de acuerdo con la matriz de riesgos. (Véase Tabla 2.3)

Tabla 2.3*Matriz de riesgo*

		Probabilidad			
Gravedad	Muy improbable	No es probable	Posible	Probable	Muy probable
Insignificante	Bajo	Bajo	Bajo	Bajo	Bajo
Menor	Bajo	Bajo	Bajo	Medio	Medio
Moderada	Bajo	Bajo	Medio	Medio	Alto
Importante	Bajo	Medio	Medio	Alto	Alto
Catastrófica	Bajo	Medio	Alto	Alto	Alto

Nota. Herramienta de análisis de riesgos para evaluar la probabilidad y la gravedad del riesgo.

El primer riesgo identificado fue económico, causado por el servicio SurveyMonkey, el cual tenía un límite máximo 500 solicitudes al día a través de su API. Estas solicitudes se utilizaban para la generación y envío de encuestas, y cada solicitud excedida tenía un costo adicional. Este riesgo fue clasificado según la matriz de riesgos:

- En la escala de gravedad se consideró como moderada, debido a que implicaba buscar una solución diferente para la generación y envío de encuestas, que requieran menos costos.
- En la escala de probabilidad se consideró como muy probable, debido a que la cantidad de usuarios que participarían realizando la encuesta sería cada vez mayor.
- Según la escala de gravedad y probabilidad se consideró como un riesgo de nivel alto.

Como solución se refactorizó la aplicación para que utilice la menor cantidad de solicitudes al día de la API de SurveyMonkey. Además, se configuró un servidor de correos con el Protocolo Simple de Transferencia de Correo (SMTP), para el envío de correos.

El segundo riesgo identificado fue de naturaleza externa, ya que el stakeholder con quien se compartía la información y se resolvían las dudas o bloqueantes contrajo el COVID-19 durante el penúltimo sprint. Este riesgo fue clasificado según la matriz de riesgos:

- En la escala de gravedad se consideró como menor, debido a que era el penúltimo sprint y estaba en una etapa de pruebas y capacitaciones con los usuarios que usarían la aplicación.
- En la escala de probabilidad se consideró como muy improbable, debido a que era muy poco probable que el stakeholder se volviera a contagiarse con el COVID-19.
- Según la escala de gravedad y de probabilidad se consideró como un riesgo de nivel bajo.

Como solución, la comunicación se realizó mediante correos con el product owner, enviando una copia al stakeholder como parte de los daily meeting.

En este nuevo reto aprendí que, para el logro de objetivos de un proyecto, no solo es necesario evaluar una solución técnica, sino evaluar el costo que conlleva implementar la solución tanto a corto plazo como a largo plazo.

Al finalizar el 2021, uno de los últimos proyectos en los que participé fue como líder del equipo de desarrollo para una empresa del sector de telecomunicaciones en Guatemala. Esta empresa decidió expandir su mercado, por lo que adquirió todos los recursos de otra empresa de telecomunicaciones del mismo país. Uno de los recursos adquiridos fue la plataforma Jira Service Desk, la cual se utilizaba para la gestión de solicitudes como mesa de ayuda. Es por ello que la nueva empresa quería realizar mejoras al proceso que ya había sido implementado dentro de la plataforma Jira Service Desk.

Para este proyecto, se utilizó la metodología Kanban debido a que los requerimientos serían solicitados de acuerdo a su complejidad y al tiempo que tomaría realizar la mejora.

Antes de iniciar el proyecto, participé en reuniones con su equipo técnico para evaluar los procesos implementados en la plataforma Jira Service Desk y comprender su necesidad de adecuar esta plataforma a su modelo actual de negocio. Además, se evaluó la necesidad de continuar utilizando los plugins instalados en la plataforma.

Mi rol dentro del proyecto era clasificar y dar seguimiento a los requerimientos ingresados por el gestor de servicios. Además, si era necesario, participaba en reuniones para entender la necesidad, validar el alcance y asignar la responsabilidad a un analista de proceso o un desarrollador, según el tipo de requerimiento.

Como parte de las actividades registradas, se realizaron mejoras a los flujos de trabajo dentro de la plataforma Jira Service Desk, tales como:

- Reducir la cantidad de roles que tenían por cada etapa en su proceso como mesa de ayuda.
- Remover procesos repetitivos como validaciones a la misma información documentada al inicio de cada etapa del proceso.
- Ejecuciones de procesos automatizados que generaban información duplicada y tareas innecesarias dentro del proyecto.
- Reducir la cantidad de automatizaciones como postfunction, validators y conditions en los flujos de trabajo para mejorar el performance de la plataforma, entre otras.

Parte de mi responsabilidad fue tener reuniones interdiarias con el cliente para poder:

- Apoyar en la toma de decisiones sobre las diferentes soluciones.
- Priorizar con el cliente las asignaciones pendientes.
- Evaluar el progreso de las asignaciones.
- Resolver impedimentos relacionados con las asignaciones.
- Limitar la sobrecarga de asignaciones.

En esta última experiencia, aprendí a evaluar diferentes alternativas para poder adecuar un sistema actual a uno nuevo, analizando junto a un equipo técnico, las mejoras que se podían realizar para lograr nuevos objetivos.

3. APRENDIZAJE CONTINUO

Mi proceso de aprendizaje inició cuando aún cursaba la universidad y aún no tenía decidido en qué área me iba a especializar. En el 2015, me interesé por la especialidad de redes y comunicaciones, por lo que decidí llevar el curso electivo de Seminario de Conectividad CCNA, que me ayudó a mejorar mis conocimientos y ponerlos en práctica en mi primer trabajo en el área de redes y comunicación de la Universidad de Lima. En este curso, aprendí a:

- Realizar configuraciones en routers y switches.
- Implementar esquemas de direccionamiento IP.
- Construir topologías de redes LAN.

Con estos conocimientos, se me asignaron nuevas responsabilidades dentro de mi área de trabajo, con las cuales pude apoyar en un proyecto de renovación de los pabellones de Administración, Psicología y Comunicaciones de la Universidad de Lima.

Luego de mi experiencia en el área de redes de comunicación, llegué a leer el libro del autor Donald Norman llamado La psicología de los objetos cotidianos, lo cual me motivó a explorar el área de desarrollo, especialmente en el diseño UX/UI. Para ello, me matriculé en un curso llamado Interacción Humano Computadora, donde me enseñaron temas como:

- Experience Design: aprendí que las soluciones no solo pueden cubrir una necesidad, sino que pueden generar emociones en los usuarios.
- User Experience: aprendí a identificar la satisfacción que un usuario tiene al interactuar con un sistema.
- Design Research: aprendí a desarrollar empatía con el usuario para poder entender la necesidad y brindar soluciones.
- Mapas de empatía y journeys: aprendí a identificar las emociones de los usuarios en cada etapa del proceso que realizaban dentro de un sistema.
- Técnicas de ideación: aprendí nuevas técnicas como Brainstorming, sketching, lightning demos, entre otros.

Debido al incremento de usuarios con dispositivos móviles, como tablets o celulares, y como complemento al conocimiento adquirido en el curso de Interacción Humano Computadora, decidí realizar el curso de Programación Móvil, donde aprendí a:

- Utilizar la herramienta Android Studio para el desarrollo de aplicaciones móviles.
- Integrar una aplicación móvil con la plataforma Firebase, para el uso de base de datos en la nube, la gestión de control de accesos con su sistema de autenticación, realizar notificaciones de tipo push, entre otras.
- Implementar buenas prácticas en el diseño de interfaces con Material Design.

En el 2018, siguiendo mi línea de carrera como desarrollador, participé en el desarrollo de una aplicación para la gestión de control de accesos de una empresa del sector logístico. Para este proyecto, era necesario desarrollar una aplicación utilizando los frameworks Angular 12 y Django 2.

Para desarrollar aplicaciones con el framework Angular, fue necesario conocer el lenguaje TypeScript. Por lo tanto, me capacité primero en los conceptos básicos en el lenguaje de programación de JavaScript y luego reforcé mis conocimientos con los conceptos básicos de TypeScript.

Además, dado que el framework de Angular se utiliza para el desarrollo de aplicaciones web, también era necesario tener conocimientos en HyperText Markup Language (HTML), para la generación de estructura de contenido, y en Cascading Style Sheets (CSS) para el diseño gráfico del documento web.

Por último, realicé un curso sobre los conceptos básicos de Angular 12, que tuvo una duración de 22 horas. Durante este curso, adquirí conocimientos sobre la creación de aplicaciones de una sola página (single-page), mediante la creación y reutilización de componentes. En Angular, cada componente está compuesto por un archivo HTML para generar la estructura, un archivo de TypeScript para agregar la lógica al componente, y un archivo CSS para añadirle el diseño.

Para el desarrollo de la aplicación con el framework Django, aprendí los fundamentos básicos del lenguaje Python, como los tipos de variables, la estructura de datos, las funciones lambda, entre otros. También adquirí conocimientos sobre cómo

configurar y crear una aplicación utilizando el framework Django, que implementaba el patrón de diseño Model-View-Controller (MVC).

De manera proactiva, leí los siguientes libros:

- Clean code (Martin, 2012): este libro me enseñó la importancia de escribir un código limpio y comprensible, lo cual facilita el mantenimiento y la realización de mejoras.
- Design patterns (Gamma, 2002): en este libro aprendí a identificar e implementar patrones de diseño como Singleton, Abstract Factory, Factory Method, Adapter, Decorator, Iterator, State y Strategy.

Como parte de mis metas, tenía el objetivo de gestionar algún proyecto como Scrum Master, por lo que decidí capacitarme en la metodología ágil Scrum. La empresa donde trabajaba utilizaba esta metodología, lo cual me motivó a tomar la capacitación. La duración de la capacitación fue de 16 horas, divididas en 8 horas de formación sobre los conceptos básicos del Scrum y 8 horas de práctica. Durante las sesiones de práctica, trabajé junto a varios participantes, divididos en grupos, donde simulamos los roles y las etapas que se habían explicado durante la capacitación.

Siguiendo con mi aprendizaje, participé en el proyecto de automatización de procesos para la gestión de ideas e iniciativas para una empresa del sector metalúrgico. Utilizamos la plataforma Jira Server en su versión 7.

Para cumplir con las necesidades del cliente, obtuve los siguientes certificados:

- Jira Server Administrator: obtuve la certificación en septiembre del 2020, entregada por Atlassian. Aprendí a configurar las funcionalidades de Jira como administrador de la aplicación, además de transformar las necesidades del cliente en soluciones dentro de la plataforma.
- Jira Project Administration: obtuve la certificación en octubre del 2020, entregada por Atlassian. Aprendí a crear y configurar proyectos dentro de la plataforma desde la perspectiva de un Scrum Master, Product Owner o Project Manager.
- Java SE 8 Programmer: obtuve la certificación en enero del 2020, entregada por Oracle University. Aprendí los fundamentos básicos de Java 8 para el desarrollo de aplicaciones orientado a objetos.

Gracias a estos certificados, pude desempeñarme como líder técnico, brindando soluciones relacionadas con la plataforma y llevando a cabo su implementación. Además, brindé apoyo al equipo de soporte en el área de mesa de ayuda al resolver errores, configurando la plataforma, creando proyectos, implementando la herramienta para las empresas, actualizando la plataforma, entre otras.

Además, fui reconocido dentro de la empresa por automatizar varios de sus procesos de ventas, en las plataformas Jira y Confluence, lo que permitió reducir en un 75% la cantidad de horas invertidas en actividades manuales.

Otro de mis mayores retos fue el proyecto de matrícula online para una empresa del sector de educación, donde requerían realizar mejoras a su plataforma de matrícula online. Para este proyecto, solicitaron que el desarrollo de su aplicación web se realizara utilizando el framework AngularJS para el front end y el framework ASP.NET 2017 para el back end.

Gran parte del conocimiento que tenía sobre el framework Angular 12 me ayudó a comprender a su predecesor, el framework AngularJS, por lo que mi curva de aprendizaje fue fácil.

Luego para empezar a desarrollar con el framework ASP.NET, realicé un curso de 12 horas sobre los conceptos básicos de programación orientada a objetos en el lenguaje C#. Aprendí sobre la estructura de datos, manejo de error, desarrollo de pruebas unitarias, entre otras.

Además, la implementación del Entity Framework en el manejo de base de datos me permitió adquirir mayor conocimiento en áreas como la creación de stored procedures, generación de triggers, consulta de datos, entre otros aspectos, dentro del sistema de gestión de base de datos SQL Server.

Al querer especializarme como desarrollador de front end, inicié aprendiendo sobre la librería React, para el desarrollo de aplicaciones web. Elegí React debido a que la curva de aprendizaje era menos compleja que la del framework Angular.

Durante este proceso, aprendí sobre la creación de componentes, los ciclos de vida de cada componente, cómo identificar cuando es necesario crear un componente como clase o como función, a utilizar Hooks creados por React y a crear Hooks personalizados, entre otros aspectos.

Al no cubrir todo el alcance total de un proyecto con la librería React, aprendí a utilizar otras librerías como React Router para la gestión de las rutas de las páginas, Redux para el manejo de los estados dentro de la aplicación, y frameworks como Next.js, que permiten crear aplicaciones web estáticas con conceptos similares a los que aprendí con React Router y Redux.

Gracias a estos conocimientos en front end, tuve la oportunidad de participar en un proyecto de Intranet para una empresa del sector de educación. En este proyecto, no solo puse en práctica mis conocimientos, sino que también recibí capacitación por parte de la empresa donde laboraba antes de iniciar el proyecto. Me capacitaron sobre la metodología de Design Thinking, la cual me enseñó a crear soluciones creativas no solo para el desarrollo de una aplicación, sino también para la identificación del producto como una marca propia.

Con este conocimiento, logré participar en las etapas de ideación y la creación del prototipo, junto al equipo interesado en el desarrollo del Intranet. Esto nos permitió alcanzar los objetivos del proyecto y entender las necesidades de los usuarios que iban a utilizar la plataforma.

Actualmente trabajo en el desarrollo de un Design System para una empresa del sector de salud. Este sistema les permite agilizar la creación de nuevas aplicaciones y mantener un diseño único en toda su marca. En este nuevo reto, apoyé en el diseño, desarrollo y la creación de pruebas unitarias de los diferentes componentes necesarios para la creación de nuevas aplicaciones, gracias a mis conocimientos en React y CSS.

En conclusión, como parte de mi experiencia profesional, puedo destacar dos áreas en las cuales adquirí gran parte de mi conocimiento:

- El área de ingeniería de software: donde entendí, debido a los primeros errores que tuve al desarrollar una aplicación, que no debe asumirse nada sin entender primero la necesidad del cliente. Aprendí a manejar siempre estándares que permita a los integrantes del equipo entender lo que se escribe como código, facilitando así la corrección de errores o la refactorización de partes del código implementado. También aprendí a controlar las excepciones generadas por la ejecución del código, lo que ayuda a controlar futuros errores. Finalmente, aprendí la creación de pruebas unitarias para mejorar la calidad del código.

- El área de ingeniería empresarial: donde aprendí dos metodologías que se utilizan en la mayoría de los proyectos en los que participé. La primera es la metodología Scrum, que permite gestionar proyectos divididos en sprints. En este enfoque, todos los participantes conocen de manera transparente el estado de las actividades, lo que ayuda a mejorar el trabajo en equipo. Además, se puede adaptar a diferentes cambios de manera inmediata durante el desarrollo del sprint. La segunda metodología es el Design Thinking,, que permite solucionar problemas de manera creativa a través de sus diferentes etapas, como la ideación de la solución, el diseño de la solución y el desarrollo de prototipos.



4. CONDUCTA ÉTICA

Desde que inicié mi experiencia profesional luego de obtener mi grado de bachiller, mi conducta es guiada por el conocimiento obtenido en el curso de Legislación y Ética Informática en la universidad, el cual fue reforzado más adelante por el enfoque del código de ética y conducta profesional de la Asociación de Maquinaria Computacional (ACM).

Durante mi experiencia laboral, un código de ética que siempre he mantenido es el de la honestidad y la confiabilidad, que se reflejan en mi forma de ser transparente y en mi disposición para proporcionar la información necesaria sobre mis capacidades para afrontar cualquier situación.

Para iniciar cada proyecto, siempre ha sido necesario evaluar mis capacidades, las cuales eran descritas de manera oral o documentada. De esta manera, generaba la confianza necesaria para formar parte de un proyecto y cumplir con el logro de los objetivos planteados.

Desde el inicio hasta el final del desarrollo de cada proyecto, hubo acuerdos de confidencialidad que era necesario conocer para lograr los objetivos planteados. Estos acuerdos incluían aspectos como el diagrama de procesos, información personal, información económica, entre otros. Dichos acuerdos no sólo evitaban la divulgación de información sensible y que no era pública sobre la empresa, sino también el uso no autorizado de sus recursos. Asimismo, se respetaba la privacidad de los usuarios externos al proyecto mediante el consentimiento de los mismos usuarios y el uso de la tecnología desarrollada para recopilar, controlar e intercambiar información personal. En este acuerdo se detalla el uso previsto de la información recopilada y el tratamiento de dicha información.

Según ACM, “el daño equivale a consecuencias negativas, especialmente cuando son significativas e injustas” (Association for Computing Machinery, párr. 1, 2018). De todos los proyectos en los que participé, fueron muy pocos los que generaron daños técnicos o daño a las relaciones interpersonales, y estos no fueron malintencionados, sino más bien involuntarios. Como equipo, todos éramos responsables y estábamos obligados a mitigar o deshacer el daño causado.

Para evitar que vuelva a suceder algún daño causado, se realizaban controles de calidad durante todo el proyecto. Estos controles se llevaban a cabo durante la etapa de desarrollo, la etapa de implementación y despliegue, y finalmente durante la etapa de prueba con los usuarios finales.

Este control de calidad fomentaba la participación de todas las partes interesadas del proyecto y mejoraba la toma de decisiones para fortalecer la confianza, y de esta manera lograr que se cumplan los objetivos de cada etapa.

Dentro del equipo de desarrollo, también se realizaban evaluaciones integrales en el desarrollo de software y el análisis de posibles impactos, de esta manera se tomaban decisiones sobre los posibles riesgos para poder evitarlos.

Parte del desarrollo consistía en implementar un sistema que garantizara que los recursos utilizados fueran controlados únicamente por personas autorizadas, con el fin de evitar cualquier uso indebido, modificación o ataque a dichos recursos. También se implementaron mecanismos de monitoreo para detectar cualquier vulnerabilidad y poder mitigarla de inmediato.

En los proyectos en los que participé como líder, mantuve la calidad de vida profesional de mi equipo sin afectar su desarrollo personal, brindándoles bienestar psicológico al mantener un buen clima laboral dentro del proyecto y bienestar físico al solicitar recursos que pudieran mejorar su ambiente de trabajo en casa.

Asimismo, como líder, facilité una buena relación entre los participantes del proyecto, reflejando confianza y buen trato entre todas las partes interesadas. De esta manera, se procuró alentar al equipo y reconocer el cumplimiento en cada etapa del proyecto.

Otro gran reto como líder fue transmitir mis conocimientos y habilidades adquiridos en cada proyecto para lograr los objetivos planteados y guiar a mi equipo, brindándoles la misma oportunidad de mejorar en sus conocimientos y habilidades.

En conclusión, tanto como miembro del equipo de desarrollo como líder técnico, el reconocimiento del código de ética profesional como parte de una ideología, me permitió:

- Identificar comportamientos que incumplían el código de ética profesional.
- Expresar el incumplimiento del código de ética profesional

- Tomar decisiones para afrontar los diversos problemas, respetando los límites.



5. LECCIONES APRENDIDAS

A lo largo de mi experiencia laboral, he participado en diferentes proyectos que me han permitido obtener diversos conocimientos, tanto a nivel personal como profesional. Sin embargo, es en las situaciones más desafiantes donde he aprendido a afrontar los obstáculos, buscar soluciones y tomar decisiones para lograr el éxito en los proyectos en los que he participado.

Durante esta larga travesía, he aprendido las siguientes lecciones:

- Como miembro del equipo de desarrollo, aprendí a participar en la planificación y a apoyar en la toma de decisiones del proyecto. Esto fue clave para lograr los objetivos planteados en el tiempo planificado, ya que demostraba mi compromiso para afrontar los desafíos planteados.
- Como líder técnico del equipo de desarrollo, aprendí a entender la necesidad del cliente para cumplir sus propios objetivos y comprender los requerimientos solicitados. Esto generaba la confianza necesaria para participar en futuros proyectos. Otro punto clave para el éxito del proyecto es la comunicación eficiente, ya sea para resolver dudas, comunicar riesgos o tomar decisiones entre todas los participantes.
- Después de graduarme de la universidad, mi experiencia tanto personal como profesional se enriqueció gracias al aprendizaje continuo. Esto me permitió tener una nueva visión de las cosas; como consecuencia, tener una actitud de progreso y estar preparado para afrontar cualquier desafío que se me presente.

Además, como lecciones aprendidas en el ámbito personal, adquirí habilidades blandas como:

- Inteligencia emocional: me permitió entender y ser autocrítico en diferentes situaciones para mejorar mis relaciones interpersonales.
- Gestión del tiempo: aprendí a priorizar las actividades dentro de mi trabajo, enfocarme en ellas y establecer límites de tiempo para evitar que afecten mi vida personal.

- **Habilidades de comunicación:** aprendí a ser asertivo y empático al interactuar con personas externas a la empresa y trabajar en conjunto para alcanzar los objetivos.
- **Manejo del estrés:** para manejar el estrés, aprendí a comunicar mis emociones en situaciones que no podía solucionar o evitar. También aprendí a pedir apoyo cuando era necesario debido a la sobrecarga de actividades y a buscar soluciones para evitar situaciones estresantes.
- **Gestión de conflictos laborales:** aprendí a detectar problemas que podrían afectar el clima laboral. Trabajando en equipo, tomábamos decisiones para prevenir y resolver conflictos tanto a nivel laboral como personal.

En conclusión, la Universidad de Lima y el aprendizaje autodidacta han sido fundamentales en la formación de mis conocimientos, tanto a nivel personal como profesional, necesarios para enfrentar los desafíos que se presentan en la realidad. Gracias a ellos, he podido destacarme como miembro o líder de un proyecto. Además, me han proporcionado la capacidad de proponer y desarrollar soluciones innovadoras que generen un impacto positivo en el logro de objetivos de un proyecto.

6. GLOSARIO DE TÉRMINOS

- **Angular**, es un framework de desarrollo construido en TypeScript que permite crear aplicaciones web escalables, basado en componentes. (Google, 2022),
- **Backlog**, es la lista de requerimientos que agregan valor, y que son seleccionados en cada sprint para ser realizado.
- **Bamboo**, es una herramienta de integración continua y despliegue automático de aplicaciones. (Atlassian, s.f.-c)
- **Bitbucket**, es un repositorio de código fuente basado en Git. (Atlassian, s.f.-a)
- **Confluence**, es una herramienta que permite a los usuarios crear y compartir documentos en un mismo espacio de trabajo. (Atlassian, s.f.-b)
- **C#**, es un lenguaje de programación orientado a objetos y con seguridad de tipos, que permite construir aplicaciones seguras y robustas que se ejecutan en .Net. (Microsoft, 2022)
- **Design system**, es una librería de software, que permite compartir componentes reutilizables para la creación de un producto de manera más rápida.
- **Django**, es un framework web basado en Python, que permite crear aplicaciones complejas utilizando el patrón Model-View-Controller (MVC). (Django Software Foundation, 2022)
- **Firebase**, es una herramienta Cloud que permite desarrollar y facilitar la creación de aplicaciones web o móviles. (Google, 2022)
- **ICMP**, es un protocolo de la capa de red para el intercambio de mensajes y control de errores en las redes IP.
- **Jira**, es una plataforma que permite la gestión de proyectos utilizando metodologías ágiles para la planificación y desarrollo de proyectos. (Atlassian, s.f.-d)

- **Object Relational Mapping (ORM)**, es un modelo de programación orientado a objetos que permite crear, leer, actualizar o eliminar datos, desde la capa de software, dentro de una base de datos. (The Apache Software Foundation, 2022)
- **Java**, es un lenguaje de programación, orientado a objetos, para el desarrollo de aplicaciones. (Oracle, 2022)
- **Kanban**, es una metodología ágil, que permite gestionar el estado actual de las tareas asignadas en el desarrollo de un proyecto. (Kanbanize, 2022)
- **Maven**, es una herramienta para construir y administrar proyectos basados en Java. (The Apache Software Foundation, 2022)
- **Plugin**, es una extensión a una determinada herramienta que permite agregar nuevas funcionalidades.
- **Product Owner**, es un rol de Scrum y es el propietario del producto que se encarga de maximizar el valor del producto.
- **React**, es una librería que permite la creación de interfaces de usuario, basado en componentes.
- **RJ45**, es una interfaz física empleada para conectar redes de computadoras mediante el uso de cableado estructurado.
- **Scrum**, es una metodología ágil para el desarrollo y mantenimiento de productos complejos. (ScrumGuides.org, 2020)
- **Sprint**, es una etapa del Scrum, donde se realizan las actividades planificadas en un tiempo determinado y que han sido seleccionadas previamente desde el backlog.
- **Stakeholder**, es un rol de Scrum conformado por los interesados en el producto, que se encargan de evaluar y proponer los ajustes necesarios para la siguiente etapa del sprint.
- **SurveyMonkey**, es una herramienta para creación y control de encuestas en línea. (Momentive, 2022)

REFERENCIAS

- Asana, Inc. (2022, 16 de marzo). *Matriz de riesgos: cómo evaluar los riesgos para lograr el éxito del proyecto (incluye ejemplos)*.
<https://asana.com/es/resources/risk-matrix-template>
- Association for Computing Machinery. (2018). *ACM Code of Ethics and Professional Conduct*. <https://www.acm.org/code-of-ethics/the-code-in-spanish>
- Atlassian. (2017, 8 de diciembre). *Create a HelloWorld plugin project*.
<https://developer.atlassian.com/server/framework/atlassian-sdk/create-a-helloworld-plugin-project/>
- Atlassian. (2022a, 30 de agosto). *Jira: advanced workflow configuration*.
<https://confluence.atlassian.com/adminjiraserver/advanced-workflow-configuration-938847443.html#Advancedworkflowconfiguration-conditions>
- Atlassian. (2022b, 12 de julio). *eazyBI Reports and Charts for Jira* (versión 3.2.0) [Software]. <https://marketplace.atlassian.com/apps/1211051/eazybi-reports-and-charts-for-jira?tab=overview&hosting=cloud>
- Atlassian. (s.f.-a). *Código y CI/CD optimizados para los equipos que usan Jira* (versión 7.12) <https://www.atlassian.com/es/software/bitbucket>
- Atlassian. (s.f.-b). *Conceptos básicos de Confluence* (versión 7.4.0) [Software] <https://www.atlassian.com/es/software/confluence/guides/get-started/confluence-overview#hosting-options>
- Atlassian. (s.f.-c). *Entrega continua desde el código hasta el despliegue* (versión 7.0) <https://www.atlassian.com/es/software/bamboo>
- Atlassian. (s.f.-d). *Una breve presentación de Jira* (versión 8.0.0) <https://www.atlassian.com/es/software/jira/guides/getting-started/overview>
- Brown, T. (2008, setiembre). Design Thinking. *Harvard Business Review*.
https://emprendedoresupa.files.wordpress.com/2010/08/p02_brown-design-thinking.pdf
- Dinngo laboratorio de innovación. (s.f.). <https://www.designthinking.es/>
- Django Software Foundation. (2022). *django*. <https://www.djangoproject.com/>
- Facebook Open Source. (s.f.). *React*. <https://es.reactjs.org/docs/getting-started.html>
- Google. (s.f.). *Angular* [Software] (versión 12.1.2). <https://angular.io/guide/what-is-angular>
- Google. (s.f.). *AngularJS* [Software]. <https://angularjs.org/>

- Google. (s.f.). *Firestore* [Software] (versión 4.1.3). <https://firebase.google.com/?hl=es>
- Kanbanize. (s.f.). *Kanban*. <https://kanbanize.com/es/recursos-de-kanban/primeros-pasos/que-es-kanban>
- Li, P. (2016). *JIRA 7 Essentials* (4.^a ed.). Packt Publishing.
- Martin, R. C. (2012). *Clean Code*. Anaya Multimedia.
- Microsoft. (2022, 18 de marzo). *A tour of the C# language*. <https://docs.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/>
- Microsoft. (s.f.). *Learn ASP.NET*. <https://dotnet.microsoft.com/en-us/learn/aspnet>
- Momentive. (s.f.). *SurveyMonkey* [Software]. <https://es.surveymonkey.com/mp/take-a-tour/>
- Mozilla Foundation. (2022, 15 de agosto). *MVC*. <https://developer.mozilla.org/es/docs/Glossary/MVC>
- Norman, D. (1990). *The design of everyday things*. Nerea.
- Oracle. (s.f.). *Developer Resources*. https://www.java.com/es/download/help/whatis_java.html
- ScrumGuides. (2020). *What is Scrum?*. <https://scrumguides.org/>
- Tempo. (2022). *Tempo*. <https://www.tempio.io/es/tempo-timesheets-seguimiento-agil-del-tiempo-para-jira>
- The Apache Software Foundation. (2022). *Apache Jmeter*. <https://jmeter.apache.org/>
- The Apache Software Foundation. (s.f.). *Introduction Apache Maven Project*. <https://maven.apache.org/what-is-maven.html>
- Ziembicki, D., Cushner, A., Rynes, A., & Tulloch, M. (2013). *Microsoft System Center: Designing Orchestrator Runbooks*. Microsoft Press

EXPERIENCIA PROFESIONAL EN EL ÁREA DE DESARROLLO DE SOFTWARE

ORIGINALITY REPORT

7 %	7 %	1 %	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	repositorio.ulima.edu.pe Internet Source	1 %
2	repositorio.neumann.edu.pe:8080 Internet Source	1 %
3	www.scribd.com Internet Source	<1 %
4	www.cobatab.edu.mx Internet Source	<1 %
5	www.coursehero.com Internet Source	<1 %
6	repositorio.ucsg.edu.ec Internet Source	<1 %
7	es.scribd.com Internet Source	<1 %
8	www.slideshare.net Internet Source	<1 %
9	seplade.michoacan.gob.mx Internet Source	<1 %