**UNIVERSIDAD TECNOLÓGICA**

**EMILIANO ZAPATA DEL ESTADO DE MORELOS**

**DIVISIÓN ACADÉMICA DE TECNOLOGÍAS DE LA INFORMACIÓN Y DISEÑO**

**SISTEMA DE GESTIÓN DE CAPACITACIONES MÓDULO DE REPORTES PARA EL DEPTO. DE DESARROLLO ACADEMICO**

**REPORTE DE ESTADÍA**

**QUE PARA OBTENER EL GRADO DE:**

**TÉCNICO SUPERIOR UNIVERSITARIO EN TECNOLOGÍAS DE LA INFORMACIÓN**

**ÁREA DESARROLLO DE SOFTWARE MULTIPLATAFORMA**

|  |  |  |
| --- | --- | --- |
| **PRESENTA:**  **JOSE EMILIO ENRIQUEZ TORRES** | | |
| ASESORA EMPRESARIAL |  | ASESOR UNIVERSITARIO |
| **DRA. MARTHA FABIOLA WENCES DIAZ** |  | **M. T. I. MAURO BAHENA CASTRO** |

**EMILIANO ZAPATA, MOR., SEPTIEMBRE DE 2023**

|  |
| --- |
| **CONTENIDO** |

Índice de figuras

Índice de tablas

Agradecimientos

Resumen

Summary

[CAPÍTULO 1. DESCRIPCIÓN DEL PROYECTO 8](#_Toc137752933)

[1.1 Datos generales de la empresa 8](#_Toc137752934)

[1.2 Antecedentes del proyecto 9](#_Toc137752935)

[1.3 Objetivo general 9](#_Toc137752936)

[1.4 Objetivos específicos 9](#_Toc137752937)

[1.5 Justificación 10](#_Toc137752938)

[1.6 Alcances 10](#_Toc137752939)

[1.7 Restricciones 11](#_Toc137752940)

[CAPÍTULO 2. MARCO DE REFERENCIA 12](#_Toc137752941)

[2.1 Conceptos, metodologías y herramientas 12](#_Toc137752942)

[2.1.1 Conceptos y principios de la programación y POO 12](#_Toc137752943)

[2.1.2 Conceptos y principios de SOLID 13](#_Toc137752944)

[2.1.3 Metodología de SCRUM 13](#_Toc137752945)

[2.1.4 Conceptos de DSM 14](#_Toc137752946)

[2.1.5 Clean Architecture 14](#_Toc137752947)

[2.1.6 Type Script 14](#_Toc137752948)

[2.1.7 Node js 14](#_Toc137752949)

[2.1.8 Vue 2 – Vue js 14](#_Toc137752950)

[2.1.9 PostgreSQL 14](#_Toc137752951)

[2.1.10 Express.js y desarrollo de API RESTful 14](#_Toc137752952)

[2.1.11 Vue Router y manejo de rutas en Vue.js 14](#_Toc137752953)

[2.1.12 JWT 14](#_Toc137752954)

[2.1.13 Entorno de trabajo: Visual Studio Code 14](#_Toc137752955)

[2.2 Propuesta de solución 15](#_Toc137752956)

[CAPÍTULO 3. DESARROLLO 17](#_Toc137752957)

[3.1 Inicio 17](#_Toc137752958)

[3.2 Planeación 17](#_Toc137752959)

[3.3 Ejecución 18](#_Toc137752960)

[3.4 Control 18](#_Toc137752961)

[3.5 Cierre 18](#_Toc137752962)

[CAPÍTULO 4. CONCLUSIONES 19](#_Toc137752963)

[4.1 Cumplimiento de objetivos 19](#_Toc137752964)

[4.2 Resultados 19](#_Toc137752965)

[4.3 Contribuciones 19](#_Toc137752966)

**REFERENCIAS**

**ANEXOS**

|  |
| --- |
| **ÍNDICE DE FIGURAS** |

[Figura 1.1Logotipo de la empresa 8](#_Toc103865183)

|  |
| --- |
| **ÍNDICE DE TABLAS** |

[Tabla 3.1 Involucrados y sus funciones 13](#_Toc103865184)

|  |
| --- |
| **AGRADECIMIENTOS** |

Texto libre escrito por el estudiante.

|  |
| --- |
| **RESUMEN** |

El texto del resumen debe redactarse para captar la atención del lector y describir brevemente lo que encontrará dentro del documento. Debe responder a las preguntas:

¿qué se hizo?, ¿cómo se hizo? y ¿qué impacto tuvo en la empresa?

De igual forma, debe incluir la estructura del documento descrita por capítulos.

|  |
| --- |
| **SUMMARY** |

Debe evitarse la traducción directa realizada por una herramienta tecnológica. Por ejemplo: Google Translator. Solicitar al alumno que sea revisado por un profesor de inglés en los horarios de asesoría.

|  |
| --- |
| DESCRIPCIÓN DEL PROYECTO |

## Datos generales de la empresa

La Universidad Tecnológica Emiliano Zapata (UTEZ) es una Universidad que ofrece planes de estudio para formar Técnicas y Técnicos Superiores Universitarios e Ingenieros e Ingenieras generando profesionistas competitivos, con altos niveles tecnológicos y sentido humanista para dar respuesta a las necesidades de los sectores social y productivo de la región y del país. El Centro de Desarrollo de Software de la Universidad Tecnológica Emiliano Zapata (CDS-UTEZ), es un centro de entrenamiento cuyo objetivo principal es la de formar estudiantes con habilidades de desarrollo y gestión de proyectos de TI, con apego a las necesidades de los clientes.



Figura 1.1Logotipo de la empresa

CDS-UTEZ se encuentra ubicada en la Av. Universidad Tecnológica No. 1. Col. Palo Escrito, C. P. 62760, Emiliano Zapata, Morelos. La UTEZ cuenta con aproximadamente 42 profesores de tiempo completo, 226 de asignatura, 3 700 estudiantes de Técnico Superior Universitario y 1500 en Licenciatura e Ingeniería.

Este proyecto se realizará bajo la tutela de M.T.I. Mauro Bahena Castro que actualmente desempeña el cargo de administrador de recursos humanos y ambiente de trabajo y como Responsable de la Administración del Proyecto Específico (RAPE), y de Ángel Yazveck Alcocer Durán como Responsable del Desarrollo (RD) del proyecto descrito en este reporte.

## Antecedentes del proyecto

Actualmente el encargado del Departamento de Desarrollo Académico de la universidad lleva la gestión de las capacitaciones por medio de archivos Excel, lo cual se hace complejo de leer, así mismo los reportes que le solicitan genera una carga de trabajo y una dedicación de horas muy grandes. Por lo tanto, al contar con un sistema que le reúna la información y genere automáticamente los reportes, podrá agilizar su proceso.

## Objetivo general

Administrar de manera ágil y eficiente las capacitaciones aplicadas a los docentes de la UTEZ, a través de la generación de reportes, utilizando una aplicación web.

## Objetivos específicos

1. Diseñar la base de datos que utilizará la aplicación y definir los métodos de acceso a datos mediante la implementación de servicios Web REST.
2. Mantener una estructura del sistema basada en Clean Architecture, donde se priorice la separación de capas y dependencia hacia el dominio para lograr un sistema modular y escalable.
3. Implementar los módulos para la gestión de las capacitaciones, docentes, áreas gestoras con el objetivo de generar reportes más precisos.
4. Implementar funciones para la carga masiva de datos de las entidades de capacitaciones y docentes mediante archivos Excel con una plantilla específica para el ágil registro y actualización de la información en el sistema.

## Justificación

El proceso actual de administración de capacitaciones mediante archivos Excel es complejo y dificulta la lectura y seguimiento de la información, la aplicación web permitirá agilizar y centralizar la gestión de las capacitaciones, mejorando la eficiencia y optimizando la gestión de la información, a su vez facilitará la gestión de reportes personalizados, brindando información específica según los criterios de filtrado establecidos permitiendo obtener datos relevantes según la necesidad.

## Alcances

* El sistema no contara con una gestión de diversos usuarios o roles, su uso está dirigido específicamente para el departamento de desarrollo académico.
* El sistema gestionara la información de los catálogos de plan de desarrollo docentes, capacitaciones parte 1 y 2, docentes, áreas gestoras y programas educativos.
* Los módulos de capacitaciones y docentes contaran con una función de registros de una carga masiva por medio de archivos Excel con una plantilla especifica.
* El sistema brindara la posibilidad de la recuperación de contraseña a través de una solicitud de recuperación de contraseña.
* Permitir a la persona responsable del proceso “Departamento de Desarrollo Académico” obtener información concreta y resumida de la base de datos.
* El sistema no gestionara ni administrara las constancias que reciban los docentes después de una capacitación.
* El sistema brindara diversas formas de filtrado de información para la generación de reportes en formato de archivo PDF.
* Brindar al usuario la posibilidad de registrar, visualizar y modificar la información de los catálogos existentes en la plataforma.
* Brindar al usuario la capacitad de generar reportes de los catálogos de Docentes y capacitaciones.

## Restricciones

* El cliente cuenta con la infraestructura necesaria para la puesta en marcha de la aplicación web.
* El cliente brindará asesoría para la resolución de dudas sobre los procesos a automatizar mediante el sistema.

|  |
| --- |
| MARCO DE REFERENCIA |

## Conceptos, metodologías y herramientas

Para el completo entendimiento del proyecto es útil comprender bajo que tecnologías, principios y procesos fue desarrollado. En base a lo recién mencionado, en este capítulo se hará exposición de los conceptos necesarios para el desarrollo inicial del proyecto.

### Conceptos y principios de la programación y POO

**Programación**

El objetivo de un programa se puede definir como un proceso que se realiza sobre ciertos datos con el fin de obtener un resultado en específico. Entonces la programación se podría entender como el momento en el que se define para qué sirve el proceso a realizar, qué datos usaras, qué resultados obtendrás y cómo se llevará a cabo ese proceso. Para poder proporcionarle las instrucciones a la computadora sobre como debe hacerlo a través de un lenguaje de programación.

**POO**

La Programación Orientada a Objetos (POO) es una técnica en la que se intenta disminuir el coste del programa aumentando la eficiencia mediante el uso de “objetos”, que se definen como el uso de entidades con atributos y formas de operar sobre los datos de forma específica mediante métodos.

**Polimorfismo**

Uno de los principios fundamentales de POO. Se entiende al polimorfismo como una cualidad que tienen los objetos de poder responder de diferente manera al mismo mensaje. Esta característica nos facilita el poder asignarles nombre a algunos métodos, ya que todas las clases derivadas de la original afectara de manera similar, pero podrán modificar su manera de responder a ella.

**Herencia**

La herencia como un principio fundamental de POO, podemos comprenderla fácilmente cuando la comparamos con la vida cotidiana cuando un padre transmite a sus hijos algo de él. En la programación la vemos como la sucesión de atributos y métodos de una clase padre o base a otra derivada de la misma haciendo que varias clases puedan tener un comportamiento similar.

**Encapsulamiento**

Este es el principio que consiste en ocultar los detalles de una clase y protegerlos mediante modificadores de acceso (public, private y protected) y solo dejar expuesta una interfaz publica para interactuar. Esto permite una mejor integridad de los datos y mejor control sobre la manipulación de los mismos.

### Clean Architecture y conceptos y principios de SOLID

**Clean Architecture**

Es un diseño de arquitectura de software propuesto por Robert C. Martin. Es un conjunto de principios y prácticas para la creación de sistemas flexibles, mantenibles siguiendo los principios de SOLID que son principios de diseño de software que promueven el modularidad, la flexibilidad, entre otras cosas.

**Principio de Responsabilidad Única (Single Responsability Principle)**

Establece que una clase debe tener una sola única responsabilidad o función en el sistema o razón para cambiar. Explicando que cada clase debe estar enfocada en realizar solo una tarea favoreciendo el modularidad y cohesión en el código. Esto porque si una clase tiene más de una responsabilidad se vuelve más difícil de entender, mantener y reutilizar.

**Principio de Abierto/Cerrado (Open/Closed Principle)**

Este principio nos indica que las entidades en el sistema como lo son las clases, módulos y funciones deben estar abiertas para cualquier extensión, pero cerradas para modificaciones. En resumen, promueve que el código este abierto a crecer sin la necesidad de modificar el código ya existente a través de nuevos módulos.

**Principio de Sustitución de Liskov (Liskov Substitution Principle)**

Este es un principio fundamental para la programación orientada a objetos, basándose en los conceptos de herencia y el polimorfismo, establece que los objetos de una clase deben poder ser sustituibles por sus clases derivadas sin que se altere el comportamiento esperado del programa. Esto significa que, si hay un objeto B que es derivado de un objeto A, el objeto B puede usarse como un objeto A sin ningún problema esto a partir de que cualquier método o atributo que exista en la clase base puede ser invocado en cualquiera de sus objetos descendientes y los resultados deberán ser coherentes y conforme a lo establecido en los contratos ya establecidos.

**Principio de Segregación de Interfaz (Interface Segregartion Principle)**

Este principio nos establece que el cliente no debe depender de interfaces que no necesita utilizar. Las interfaces deberán ser lo mas especificas posibles, intentando ofrecer solo los métodos y funcionalidades necesarias promoviendo la idea de dividir las interfaces que sean muy grandes y poco cohesivas en unas mas pequeñas evitando la sobrecarga y dependencia innecesaria de funcionalidades que no son necesarias o utilizadas.

**Principio de Inversión de Dependencia (Dependency Invercion Principle)**

El principio de DIP nos establece que los módulos de alto nivel no deberán ser dependientes de los de bajo nivel ni viceversa, ambos deben de depender de abstracciones, promoviendo las clases abstractas e interfaces en lugar de otras implementaciones concretas. Además, las abstracciones no deberán de depender de detalles específicos sino al revés provocando una arquitectura más sólida, flexible y facilitando la introducción de nuevos componentes, la reutilización de código y las pruebas al sistema.

### Metodología de Scrum

**Marco de trabajo Ágil**

Scrum es una estructura para el trabajo ágil, en la actualidad es altamente empleado en el desarrollo de software, sistemas y proyectos de gestión. Se centra principalmente en la entrega iterativa, colaboración entre los integrantes del equipo y una adaptación continúa, permitiendo responder de manera mas flexible a cambios en requisitos y entregas de avance de una forma incremental.

Scrum se basa en tres pilares fundamentales:

1. Transparencia: nos dice que todo tipo de información que se vaya generando sobre el proyecto sea clara, visible y comprensible para todos los miembros del equipo, algo que fomenta la confianza y facilita la toma de decisiones.

2. Inspección: Los equipos deben de inspeccionar su trabajo, el progreso y los resultados obtenidos. Mediante la revisión continua de la calidad de un producto, se pueden identificar mas oportunidades de mejora y poder tomar medidas para las desviaciones.

3. Adaptación: se fomenta la adaptación continua gracias a los aprendizajes adquiridos durante el desarrollo del proyecto para poder dar respuesta a cualquier tipo de cambio que surja. Esto es de suma importancia, ya que permite que el equipo ajustarse ágilmente a algún nuevo lineamiento.

**Roles y responsabilidades**

**Artefactos y eventos**

### Conceptos de DSM

### Type Script

### Node js

### Vue 2 – Vue js

### PostgreSQL

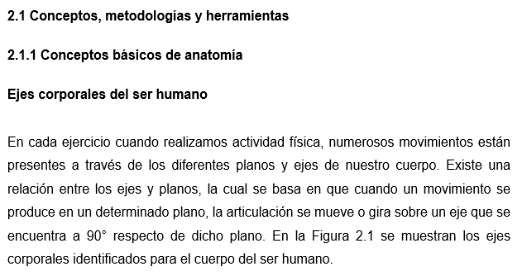
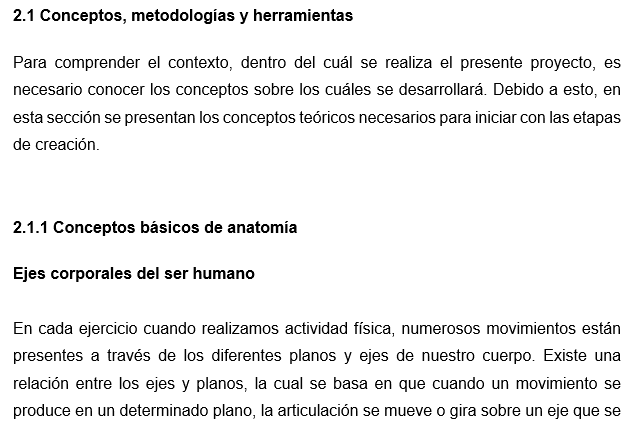
### Express.js y desarrollo de API RESTful

### Vue Router y manejo de rutas en Vue.js

### JWT

### Entorno de trabajo: Visual Studio Code

En este capítulo, es necesario que bajo los títulos de nivel 2 (los que tienen dos números, como 2.1), se coloque una introducción o descripción breve del tema. Con este se evita tener múltiples títulos juntos y da mejor compresión del texto.



Esta sección deberá contener subtítulos que agrupen conceptos, metodologías y herramientas tecnológicas necesarios para el desarrollo del proyecto. Es necesario colocar al menos tres opciones por cada subtítulo que se incluya. Estos subtítulos se pueden considerar como categorías. Adicionalmente, si existiesen trabajos o productos comerciales relacionados con el proyecto, se pueden colocar.

En esta sección no se admiten logotipos. Si se desea colocar imágenes, deberán ser necesarias y significativas para la explicación del texto redactado, deben poder distinguirse los elementos que contiene (texto o formas) y no repetirse en algún otro lugar del documento. Recuerda que todo las imágenes y tablas deberán estar referenciadas en el texto y, si no son las creaste tú, deberás incluir la cita del autor de dónde la obtuviste. Por ejemplo, si fueses a realizar una aplicación de realidad aumentada para entrenamiento de cirujanos, esta sección podría quedar estructurada como:

2.1.1 Conceptos básicos de anatomía

Ejes corporales del ser humano

…

2.1.2 Software para modelado 3D

3DS Max

Maya

Blender

2.1.3 Herramientas de desarrollo para Realidad Aumentada

Vuforia

Como puedes observar, los títulos de 4º nivel no llevan numeración, pero se siguen considerando como títulos para efectos de formato.

## Propuesta de solución

Esta sección debe describir la propuesta de solución planteada fundamentando la elección de herramientas, tecnologías y métodos con base en lo analizado en la sección anterior. Debe incluir la viabilidad del proyecto.

El capítulo 2 deberá tener un mínimo 10 hojas y un máximo 15. De igual forma, la sección 2.1 deberá contener como mínimo 5 referencias para TSU y 10 para Ingeniería.

|  |
| --- |
| DESARROLLO |

## Inicio

Esta sección debe describir las acciones realizadas para iniciar con el proyecto. Generalmente describe la reunión inicial con el asesor institucional, con la que se genera el Project charter o el Acta constitutiva del proyecto.

Tabla 3.1 Involucrados y sus funciones

|  |  |  |
| --- | --- | --- |
| **INVOLUCRADOS EN EL PROYECTO** | | |
| **Nombre de la persona** | **Función** | |
| Nombre del sponsor | | Sponsor  *Interno* |
| Nombre del cliente | | Cliente  *Externo* |
| Nombre del asesor | | Asesor  *Interno* |
| Nombre del administrador del proyecto | | Administrador del proyecto  *Interno* |

## Planeación

Esta sección debe describir las acciones realizadas para planificar el proyecto y deberá contener como mínimo un documento que defina los requerimientos del proyecto y un cronograma de actividades (o diagrama de Gantt) que indique tiempo de duración y entregables. Adicionalmente, se podrán añadir documentos que defina tu asesor universitario de acuerdo con el tipo de proyecto específico.

## Ejecución

En esta sección se deben describir, paso a paso, las acciones realizadas para desarrollar el proyecto. Estas acciones deben tener un orden cronológico coherente. Similar a un tutorial.

Normalmente se escribe en pasado, distinguiendo los elementos del estado actual del proyecto (escrito en presente) y aquellos que se tendrán como resultado de una acción futura (escritos en futuro).

## Control

Esta sección debe describir las acciones realizadas para llevar el control del proyecto. Los elementos por colocar pueden variar dependiendo del proyecto. Se pueden colocar elementos como: descripción de reuniones periódicas para el seguimiento del proyecto, seguimiento del proyecto considerando la planeación de actividades en comparación con el avance real, bitácora de cambios en el proyecto o seguimiento de riesgos.

## Cierre

Esta sección debe describir las acciones realizadas para concluir el proyecto de manera formal. Debe incluir al menos las acciones realizadas para entregar el proyecto (reunión), la descripción de los entregables del proyecto al cliente y la aceptación del cliente (o carta de liberación).

|  |
| --- |
| CONCLUSIONES |

## Cumplimiento de objetivos

Se debe redactar una descripción que indique cómo se cumplieron los objetivos (generalmente los específicos) y por qué se dice que se cumplieron o por qué no se cumplieron. Para ambos casos, se debe argumentar la razón.

## Resultados

Se deben describir los entregables finales, probados y en operación, así como los beneficios obtenidos por la empresa como resultado del proyecto.

## Contribuciones

Se deben describir las aportaciones a la empresa. Generalmente, no medibles. Por ejemplo: se utilizó una metodología nueva para el desarrollo. ¿Qué se utiliza ahora en la empresa que no se usaba antes?

De igual forma, se debe describir como benefició el proyecto a la formación profesional

del alumno. NO ESCRIBIR EN PRIMERA PERSONA. Al igual que el resto del documento, se debe escribir en tercera persona.

|  |
| --- |
| **REFERENCIAS** |

**No hay ninguna fuente en el documento actual.**

|  |
| --- |
| **ANEXOS** |

**ANEXO A. NOMBRE DEL ANEXO**

Los anexos, además de tener una letra, deberán indicar un nombre para poder identificarlos. No son títulos, por lo que no aparecerán en el índice.

**ANEXO B. NOMBRE DEL ANEXO**

**Estilos del formato de reporte**

Este archivo incluye 5 estilos definidos para títulos y texto del contenido, que puedes encontrar en la banda **Inicio** de Word, y son:

* **Título 1**: para los títulos de capítulo.
* **Título 2**: para los títulos de segundo nivel. Por ejemplo: 1.2
* **Título 3**: para los títulos de tercer nivel. Por ejemplo: 1.2.1
* **Descripción**: para los títulos de figuras y tablas.
* **Normal**: para el texto del contenido.



Todo el texto del documento debe tener aplicado al menos uno de estos estilos, según corresponda.

Se deben utilizar los estilos para que, al actualizar los índices de contenido, se adopten adecuadamente los textos colocados como títulos y se indiquen las páginas donde se encuentran.

Los títulos del índice de contenido deben ser los únicos elementos numerados.

**Portada**

Toda la portada deberá estar escrita en mayúsculas. Se debe elegir la palabra que defina correctamente a los asesores empresarial y universitario (asesor o asesora, universitario o universitaria); es decir, el texto colocado en rojo deberá sustituirse según corresponda.



Se debe colocar correctamente el nombre del programa educativo, considerando dos escenarios:

* Los nombres de programas que no tienen área (Ingenierías) se colocan en tamaño 16.
* Los nombres de programas que tienen área (TSU) deben seguir la regla anterior y, además, colocar el área en tamaño 14.

Es necesario que la portada tenga centrados tanto el nombre del proyecto como el grado. Esto implica que no solo debe estar centrado horizontalmente, sino también verticalmente en el espacio que corresponde a cada texto (área blanca para el nombre del proyecto y área azul para grado).

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

El lugar y fecha para el reporte se encuentra en la parte inferior de la portada. Se debe verificar que el mes que se indica sea un mes posterior a la finalización de la estadía.



**Formato del documento**

Se deberá respetar el formato del documento actual. Si por algún motivo se llegase a modificar, es necesario revisar los siguiente:

* Los márgenes del documento serán de 3cm. a la izquierda, 2.5 cm. a la derecha, superior e inferior.
* No se deberá incluir ninguna leyenda (ni líneas, ni títulos de trabajos, ni nombre del alumno) de encabezado.
* El pie de página solo deberá contener el número de la página fuente arial 10, alineado a la derecha.

**Cantidad de páginas**

* Las páginas del documento empiezan a contar después de la portada, que se considera la página 0, pero el número de página se hace visible hasta el Capítulo 1.
* La cantidad de páginas que deberá contener el reporte de estadía será de 40 como mínimo, las cuáles **se empiezan a contar a partir de la** **primera página del Capítulo 1** y **hasta la última página del capítulo 4**. No se consideran portada, índices, agradecimientos, resumen, referencias o anexos.
* Por ejemplo: un reporte que **inicia en la página 8** deberá **termina**r, al menos, **en la página 47**.

**Saltos de línea**

Se deberán considerar las siguientes reglas para los saltos de línea o renglones vacíos entre párrafos, títulos, imágenes y tablas:

* Se coloca un salto de línea entre título y párrafo; y entre párrafo y párrafo.
* Las imágenes y las tablas se consideran parte del texto o un párrafo. Si una imagen se encuentra con una tabla, o viceversa, se coloca un salto de línea entre ellas.
* Se colocarán dos saltos de línea entre párrafo y título siguiente.

Cuando un título queda solo al final de una página, existen dos opciones:

* Si el espacio restante es un salto de línea, se puede insertar un salto de página antes del título para moverlo a la siguiente página.
* Si el espacio restante es más de un salto de línea, se debe complementar la redacción del texto previo para evitar que el título quede solo al final de la página.

****

**Redacción y ortografía**

El documento debe estar escrito en tercera persona y evitar el uso del verbo ser seguido de un adjetivo. Por ejemplo:

* **No** **escribir**: Fue necesario, fue requerido, fue desarrollado, etc.
* **Utilizar**: Se necesitó, se requirió, se desarrolló.

Se debe evitar el uso de la palabra nosotros. Por ejemplo:

* **No escribir**: Tuvimos, realizamos, obtendremos, etc.
* **Utilizar**: Se tuvo, se realizó, se obtendrá, etc.

Es necesario revisar acentuación del documento (F7 con Word) y prestar especial atención en verbos escritos en pasado y futuro.

Se debe revisar que el texto este redactado procurando utilizar enunciados cortos. Generalmente se hace un uso excesivo de comas en lugares erróneos. Para verificar la necesidad de un punto, solicitar:

***Lee los párrafos en voz alta. Si durante la lectura te falta el aire para terminar, es posible evidencia de que requieres un punto o punto y coma.***

**Numeración y viñetas**

Se prefieren las viñetas sobre la numeración, salvo cuando sea necesario indicar un orden. **NO** se deben colocar saltos de línea entre numeración o viñetas. **Esto se debe observar a detalle, ya que se abusa de ello** con el fin de obtener hojas adicionales.

Se debe colocar un salto de línea antes de iniciar el listado con las viñetas.

**Referencias**

Las referencias se deben incluir en el texto y, en la medida de lo posible, seguir el formato APA utilizado por Microsoft Word. Para ello, se deben utilizar las herramientas para la gestión de referencias que proporciona el editor.

Se deberá evitar el plagio. Evitar el texto copiado y pegado. Si no hubiese otra opción, referenciarlo correctamente. Es preferible parafrasear o escribir con palabras propias.

**Referencia parafraseada:** comunica el mensaje del autor original con palabras propias y conservando el sentido en cuanto a organización, continuidad y coherencia. Generalmente, se inicia con una frase introductoria (así como, por tanto, de acuerdo con, según, etc.)

*De acuerdo con (Salazar, 2018), un programa computacional aplicado (PCA) se encarga de resolver un problema trivial en el campo...*

**Referencia textual:** es una cita fiel del texto original de un autor, palabra por palabra. El texto debe ponerse entre comillas, acompañados de la referencia.

*Para comprender los ejes corporales del cuerpo humano, "se han concebido planos imaginarios de referencia que atraviesan el cuerpo de forma que unos son perpendiculares a otros" (Palastanga, Field & Soames, 2000).*

**Las referencias deben tomarse de sitios confiables** y debe priorizarse la referencia de elementos obtenidos de libros, enciclopedias, artículos científicos, artículos de revistas académicas, entre otros.

Los documentos o sitios Web deben ser confiables, de entidades reconocidas como expertos o de renombre en el área.

Si se trata de una **entrada de blog**, debe tratarse de una persona reconocida en el campo (experto, no necesariamente popular). Por ejemplo: Richard Stallman, Bruno Munari, entre otros.

**Imágenes y tablas**

Si se añade una imagen es porque:

* Ayuda a la compresión del texto y debe describirse.
* El lector puede distinguir el texto y formas que incluye la imagen.
* Es necesaria y no se repite en algún otro lugar del documento.

El **tamaño de las imágenes** en el reporte quedará **a consideración del asesor universitario**, mismo que validará que sean necesarias y no se abuse de ellas para ocupar espacio.

Es válido colocar dos imágenes seguidas siempre que sea necesario para mantener el correcto formato del documento. **Se debe verificar que no se abuse de esto.**

Las imágenes y tablas NO deben colocarse en cuadros de texto o con diseño flotante (que pueda arrastrarse con el mouse sobre el documento). Las imágenes deben colocarse en línea con el texto.

Las imágenes deben estar referenciadas en el texto. Si se añade una imagen es porque ayuda a la compresión del texto y debe describirse. Por ejemplo, para hacer referencia a la figura con título Figura 2.1 en el texto descriptivo, se deberá hacer como:

“… en la Figura 2.1 se puede observar el funcionamiento paso a paso del algoritmo de búsqueda Bubble Sort para un arreglo de datos numéricos aleatorios.”

El rótulo y número, en el título de las imágenes, debe colocarse en texto normal (sin cursivas o negritas). Por ejemplo: Figura 2.1 Servidor de base de datos. El título de las tablas se coloca en la parte superior del encabezado, también en texto normal (sin cursivas o negritas).

Los títulos de imágenes o tablas se deben colocar sin punto.

Si una imagen no fue de creación propia, es necesario agregar la referencia al título de las imágenes. Por ejemplo:

Gráfico, Gráfico de líneas

Descripción generada automáticamente

No debe existir algún salto de línea entre la imagen y su título de imagen. Deben cortarse los espacios en blanco alrededor de la imagen.



Cuando un título queda solo al final de una página, existen dos opciones:

* Si el espacio restante es un salto de línea, se puede insertar un salto de página antes del título para moverlo a la siguiente página.
* Si el espacio restante es más de un salto de línea, se debe complementar la redacción del texto previo para evitar que el título quede solo al final de la página. Por ejemplo:

