**UNIVERSIDAD TECNOLÓGICA**

**EMILIANO ZAPATA DEL ESTADO DE MORELOS**

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

**MODULO EVALUACIÓN DOCENTE PARA EL SISTEMA INTEGRAL UNIVERSIDAD DE CIENCIAS JURIDICAS**

**REPORTE DE ESTADÍA**

**QUE PARA OBTENER EL GRADO DE:**

**NOMBRE DEL PROGRAMA EDUCATIVO**

**ÁREA DEL PROGRAMA (SI APLICA)**

|  |  |  |
| --- | --- | --- |
| **PRESENTA:**  **NOE MARTINEZ FLORES** | | |
| ASESORA EMPRESARIAL |  | ASESOR UNIVERSITARIO |
| **DRA. MARTHA FABIOLA WENCES DÍAZ** |  | **I.T.I ERICK MIRELES MERCHANT** |

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

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

Índice de figuras

Índice de tablas

Agradecimientos

Resumen

Summary

[CAPÍTULO 1. DESCRIPCIÓN DEL PROYECTO 8](#_Toc103865162)

[1.1 Datos generales de la empresa 8](#_Toc103865163)

[1.2 Antecedentes del proyecto 9](#_Toc103865164)

[1.3 Objetivo general 9](#_Toc103865165)

[1.4 Objetivos específicos 10](#_Toc103865166)

[1.5 Justificación 10](#_Toc103865167)

[1.6 Alcances 11](#_Toc103865168)

[1.7 Restricciones 11](#_Toc103865169)

[CAPÍTULO 2. MARCO DE REFERENCIA 12](#_Toc103865170)

[2.1 Conceptos, metodologías y herramientas 12](#_Toc103865171)

[2.2 Propuesta de solución 13](#_Toc103865172)

[CAPÍTULO 3. DESARROLLO 14](#_Toc103865173)

[3.1 Inicio 14](#_Toc103865174)

[3.2 Planeación 14](#_Toc103865175)

[3.3 Ejecución 15](#_Toc103865176)

[3.4 Control 15](#_Toc103865177)

[3.5 Cierre 15](#_Toc103865178)

[CAPÍTULO 4. CONCLUSIONES 16](#_Toc103865179)

[4.1 Cumplimiento de objetivos 16](#_Toc103865180)

[4.2 Resultados 16](#_Toc103865181)

[4.3 Contribuciones 16](#_Toc103865182)

**REFERENCIAS**

**ANEXOS**

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

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

|  |
| --- |
| **ÍNDICE DE TABLAS** |

[Tabla 3.1 Involucrados y sus funciones 14](#_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

El centro de desarrollo de software es un centro de entrenamiento para personas y/o estudiantes de nivel superior enfocados en la carrera de Tecnologías de la Información y Comunicación, donde los participantes no solo adquieran conocimientos teóricos, sino que también obtengan una valiosa experiencia práctica en proyectos reales centrados en el desarrollo de software.



*Figura 1.1 Logotipo de la empresa*

CDS se encuentra dentro de las instalaciones de la UTEZ ubicada en avenida Universidad Tecnológica 1,Palo Escrito, Emiliano Zapata Morelos, C.P 62765 frente a la preparatoria CECyTE, cuenta aproximadamente con 500 empleados los cuales son distribuidos en ciertas áreas o carreras que son distribuidas a lo largo de toda la universidad.  
   
Este proyecto se realizará bajo la tutela de Erick Mireles Merchant, docente de   
la Universidad Tecnológica de Emiliano Zapata (UTEZ) que actualmente desempeña   
el cargo de docente y del proceso de Gestión de Proyectos dentro del CDS-UTEZ.

## Antecedentes del proyecto

Con el constante crecimiento de la Universidad de Ciencias Jurídicas, la gestión de estudiantes, docentes y procesos administrativos ha generado una demanda cada vez más compleja y robusta. El sistema existente, aunque funcional en ciertos aspectos, ha evidenciado limitaciones que no cumplen adecuadamente con las necesidades actuales de la institución.

Se ha decidido buscar el apoyo del Centro de Desarrollo de Software (CDS) para llevar a cabo la creación de una aplicación híbrida que aborde de manera específica y efectiva los desafíos presentes en la gestión académica y administrativa. Aunque actualmente existe un sistema en funcionamiento, su capacidad se ha quedado corta y no logra satisfacer completamente las demandas y expectativas de la universidad.

## Objetivo general

Administrar el proceso de evaluación docente en la Universidad de Ciencias Jurídicas a través de una aplicación híbrida, con el propósito de proporcionar a todos los docentes retroalimentación directa de los alumnos, permitiéndoles identificar y mejorar sus puntos de mejora o fortalezas en el ámbito educativo.

## Objetivos específicos

1. Diseñar los modelos necesarios dentro de la base de datos para almacenar la información de la evaluación.
2. Diseñar e implementar los servicios API para consulta, registro y cambio de estado de aquellos datos relacionados a la evaluación docente.
3. Implementar los privilegios que tendrán los usuarios para poder realizar los procesos que conllevan a la creación y modificación de la evaluación docente.
4. Diseñar una interfaz intuitiva y amigable para el usuario final, garantizando la accesibilidad y facilidad de uso para los estudiantes y los encargados de hacer la evaluación docente.
5. Implementar las funcionalidades para la gestión de secciones, preguntas y consulta de las respuestas de la evaluación docente.
6. Implementar pruebas funcionales y probar la aplicación en un servidor.

## Justificación

El desarrollo e implementación de un Módulo de Evaluación Docente en la Universidad de Ciencias Jurídicas es esencial para optimizar el proceso de evaluación del desempeño docente. Este módulo permitirá a los estudiantes expresar sus opiniones y evaluaciones de manera estructurada, recopilando datos significativos y proporcionando retroalimentación detallada.

La introducción de este sistema contribuirá a la transparencia y rendición de cuentas, promoviendo una cultura de mejora continua en la universidad. Además, alineado con las mejores prácticas educativas, el Módulo de Evaluación Docente no solo cumplirá con estándares de calidad, sino que también fortalecerá la reputación de la institución al demostrar su compromiso con la excelencia académica y la satisfacción estudiantil.

## Alcances

* Generación del manual de usuario para las respectivas funcionalidades de: secciones, preguntas, detalles de la evaluación docente, evaluación docente.
* Pruebas al sistema.
* Redacción del DFR.
* Generación de un plan de pruebas.
* Generación de análisis y diseño del módulo.
* El desarrollo del proyecto no incluirá el desarrollo de una aplicación móvil.

## Restricciones

* El proyecto debe cumplir con reglas de usabilidad para facilitar la experiencia del usuario final.
* Las reuniones son echas de manera remota, por video llamada por medio de plataformas como Google Meet, esto conlleva a ideas no completas o casos donde la información de un receptor a otro no sea lo suficientemente clara para el avance del proyecto.

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

## Conceptos, metodologías y herramientas

Para comprender el contexto dentro del proyecto, es necesario conocer los conceptos acerca de las tecnologías, lenguajes, frameworks, entornos de trabajo, entre otros. Para ello se realizó una investigación y posterior explicación de diferentes conceptos relacionados al proyecto.

### Lenguajes de Programación

**Java**

El lenguaje de programación de Java fue seleccionado gracias a su alta madurez en el desarrollo de software, cuenta con una gran escalabilidad para proyectos medianos y grandes, cuenta con variables con un alto tipado estricto, lo cual lo hace ideal para el manejo de datos por parte del backend en el desarrollo de este proyecto. Además, fomenta el uso de buenas prácticas, así como de la arquitectura del proyecto.

**JavaScript**

Es un lenguaje altamente compatible para utilizar dentro de los navegadores web de hoy en día, además de ello que cuenta con una extensa cantidad de framewroks, librerías creadas por la comunidad de desarrolladores, soporte anual gracias a los módulos del ECMAScript. Junto a que, gracias a la modificación y completo control del DOM (Docuemnt Object Model) se puede crear experiencias interactivas entre el usuario dinámicas y completamente personalizables.

**TypeScript**

TypeScript, como un superconjunto de JavaScript, aporta una valiosa capa de tipado estático y funciones avanzadas al lenguaje. La introducción del tipado estático proporciona claridad y confianza al código, ya que los tipos de datos pueden ser definidos y verificados durante la fase de compilación, eliminando potenciales errores de tipo en tiempo de ejecución. Esta característica no solo facilita la comprensión del código, sino que también mejora significativamente la seguridad y robustez del desarrollo. El tipado estático de TypeScript no solo contribuye a la claridad, sino que también reduce la mutación de variables, fomentando buenas prácticas de programación y facilitando el mantenimiento del código a lo largo del tiempo.

**SQL**

El lenguaje SQL, Structured Query Language, es una elección sólida para proyectos de cualquier escala debido a su capacidad integral para gestionar datos en bases de datos. Este lenguaje permite realizar operaciones fundamentales como el almacenamiento, consulta, modificación y eliminación de datos. Además de ello, el uso de tablas garantiza una estructura organizada permitiendo presentar la información de manera clara y lógica. La capacidad de establecer relaciones entre tablas gracias a las llaves foráneas facilita el uso de consultas o instrucciones complejas.

### Herramientas de prototipado

Parte de los prototipos son una pieza fundamental para llevar a cabo dicho proceso,

ya que con esto se tiene una mejor visión de los conceptos principales y básicos del

proyecto, con esto se puede compartir las opiniones con el equipo y ver cuál opción

es la más idónea.

Según (Galina, 2021) aparte fabricar un prototipo puede aportar varios beneficios

como:

* Evitar hacer una gran inversión al principio en un producto que no termina de gustar a los usuarios.
* Reduce el riesgo, ya que, si alguna función del producto no guste, se deja de implementarla antes de la fabricación.
* Se puede mejorar el producto según la aceptación que ha tenido entre los clientes.

**Figma**

Según (Blandino, 2023) Figma bien permite diseñar prototipos web o móvil, presentaciones, es muy útil porque facilita el trabajo, ya que es basada en navegador 14 y puedes interactuar con las personas de tu equipo ya que es una herramienta colaborativa.

### Frameworks utilizados para frontend

**Vue JS**

Según (Jesus, 2022) permite una interfaz de usuario más reactiva y receptiva. Además de ello, El objetivo principal de Vue JS es simplificar el desarrollo web y los sitios web. Gracias a su pequeño tamaño de compilación, su facilidad de uso, aprendizaje, y rendimiento.

**Vue Boostrap**

Vue Bootstrap combina la potencia del framework de desarrollo frontend Vue.js con la versatilidad y estilos predefinidos de Bootstrap. Vue JS proporciona un enfoque reactivo para las interfaces de usuario, mientras que Bootstrap ofrece un conjunto de estilos y componentes que agilizan el diseño y la apariencia de las aplicaciones.

### Framework utilizados para backend

**Spring Boot**

Según (Fernández, 2022) Estas son las características más destacadas de este framework:

* Permite crear todo tipo de aplicaciones en el lado del back-end de forma independiente.
* Facilita el trabajo con otras herramientas como Tomcat, Jetty o Undertow. Lo hace directamente, sin necesidad de implementar archivos específicos para ello.
* Simplifica las dependencias para mejorar la configuración final del proyecto que se desarrolla con Spring Boot.
* Se trata de un framework que se configura de manera simple y es compatible con bibliotecas de terceros.
* Facilita la creación de listas, controles de estado y mejora la configuración externa para el desarrollo de aplicaciones.

Gracias a estas características, Spring Boot facilita la creación de todo tipo de aplicaciones basadas en él de manera independiente con el mínimo esfuerzo por parte de los desarrolladores.

### Herramientas para la gestion de base de datos

**PostgreSQL**

PostgreSQL es un sistema de gestión de bases de datos relacional (RDBMS) que destaca por su robustez, flexibilidad y capacidad para manejar cargas de trabajo variadas.

Segun (Borges, 2015) estas son algunas de sus mas grandes caracteristicas:

* Multiplataforma: La versatilidad de PostgreSQL es evidente en su capacidad para funcionar en diversas plataformas, incluyendo sistemas operativos Unix y Windows. Esta característica elimina las preocupaciones relacionadas con la compatibilidad y permite su uso tanto localmente como a través de conexiones remotas, ofreciendo flexibilidad para adaptarse a diferentes entornos.
* Alto Volumen y Rendimiento: PostgreSQL destaca en el manejo de bases de datos con alto volumen de información gracias a su eficiente método de Control de Concurrencias Multiversión (MVCC). Este enfoque permite lecturas y escrituras simultáneas sin bloquear operaciones, mejorando significativamente el rendimiento en entornos con movimientos intensivos en la base de datos.
* Facilidad de Manejo: El principal administrador de bases de datos de PostgreSQL, PgAdmin, ofrece una interfaz sencilla y fácil de manejar. Ideal para aquellos que están comenzando en el mundo de las bases de datos o necesitan una solución para sus proyectos, PgAdmin simplifica tareas complejas mediante simples clics. Además, proporciona opciones avanzadas para usuarios experimentados que prefieren gestionar códigos específicos de manera directa.

**DataGrip**

DataGrip es una herramienta diseñada para gestionar sistemas de bases de datos SQL en entornos multiplataforma, siendo compatible con Windows, macOS y Linux. Este IDE aborda específicamente el lenguaje SQL y facilita la interacción con bases de datos. DataGrip es completamente compatible con diversas bases de datos, incluyendo PostgreSQL, además de MySQL a partir de la versión 5.1. Su funcionalidad abarca sistemas operativos como Windows, macOS y Linux. (JetBrains, 2024)

### Herramientas de desarrollo

**VSCode**

VSCode cuenta con una interfaz intuitiva y con una capacidad ampliamente compatible para aceptar y ejcutar varios lenguajes de programación, lo que lo convierte en una herramienta valiosa para desarrolladores de diferentes tecnologías. Además, su amplio conjunto de extensiones permite personalizar el entorno de desarrollo según las necesidades específicas del proyecto y del desarrollador.

La integración de Git directamente en el IDE facilita el control de versiones y la colaboración en equipo. También ofrece características como IntelliSense, depuración integrada, resaltado de sintaxis y una poderosa capacidad de búsqueda, lo que mejora la productividad y la eficiencia durante el desarrollo.

**Intellij**

Segun (Trucoteam, 2024) Es un IDE que ofrece una amplia variedad de ventajas, entre ellas se pueden econtrar:

Mejora de la productividad: Ofrece un conjunto de herramientas que te ayudarán a optimizar tu tiempo al escribir código y revisar errores, reduciendo las horas de trabajo necesarias para programar una aplicación.

* Entorno de trabajo flexible: IntelliJ IDEA se puede personalizar fácilmente para adaptarse a tus necesidades y a la arquitectura de tu aplicación. Además, puedes cambiar entre una amplia variedad de lenguajes de manera rápida y sencilla, permitiéndote aprovechar las características de diferentes lenguajes sin problemas.
* Depuración intuitiva: El IDE viene con una amplia variedad de herramientas para ayudarte a encontrar y corregir errores rápidamente. Es fácil realizar pruebas de rendimiento profundo con IntelliJ IDEA, y puedes hacer uso del potente analizador de líneas de código para rastrear problemas.
* Soporte para múltiples plataformas: IntelliJ IDEA es compatible con Java, Kotlin, Groovy y otros lenguajes de programación más recientes, y también ofrece una solución óptima para plataformas como Android, Spring, Maven y Gradle.
* Más de 100 integraciones disponibles: El IDE te proporciona una amplia gama de integraciones y contenido adicional para ayudarte a realizar tus tareas de manera más rápida. Estas herramientas incluyen una construcción de prueba JUnit, actualización del depurador, herramienta analizadora de reglas de codificación, control de versiones y mucho más.

**GIT**

Git es un sistema de control de versiones distribuido, forma parte de los equipos de desarrollo donde gestionan y colaboran en el código fuente. Git no solo es usado únicamente por desarrolladores o creadores de código, también es utilizado por profesionales de distintos enfoques como Diseñadores digitales ya que su uso no es únicamente para archivos de programación, si no que involucra cualquier tipo de archivo electrónico. Permite un seguimiento preciso, facilita la creación y fusión de ramas, optimiza la colaboración entre los integrantes de un equipo o de manera individual, y garantiza un historial detallado de los cambios realizados a los archivos afectados o modificados.

### Arquitectura

**Clean Arquitecture**

Clean Architecture busca proporcionar una estructura clara y sostenible para las aplicaciones. Estas aplicaciones pueden ser escalables y sostenibles a lo largo del tiempo.

Según (Rosal, 2023) La arquitectura limpia es ideal cuando tienes un proyecto a largo plazo, que perdure en el tiempo, que los test sean realizados con facilidad y los cambios no sean un desafío de implementar en cualquier parte de la vida del proyecto. Algunos de los beneficios de esta clase de arquitectura son los siguientes elementos:

* Implementación Inmediata con lenguajes de programación como Java, .Net, NodeJS, etc.
* El foco primario del proyecto se centra en la lógica del dominio o reglas de negocio.
* Esta arquitectura permite realizar cambios importantes en la aplicación, sin grandes impactos:
  + Podrías cambiar el framework utilizado en caso de ser necesario, ya que está todo desacoplado.
  + Podrías, además, cambiar la base de datos que uses o agregar alguna otra si la necesitas.
* Tienes la oportunidad de testear de manera rápida y fácil.
* Crearás un producto sólido, de calidad y escalable.

### Metodologías empleadas

**SCRUM**

Según (Schwaber & Sutherland, 2020) Scrum involucra a grupos de personas que colectivamente tienen todas las habilidades y experiencia para hacer el trabajo y compartir o adquirir tales habilidades según sea necesario.

El equipo de Scrum se compromete a lograr sus objetivos y apoyarse mutuamente. Su enfoque principal es el trabajo del Sprint para hacer el mejor progreso posible hacia estos objetivos. El equipo de Scrum y sus partes interesadas están abiertos sobre el trabajo y los desafíos. Los miembros del equipo de Scrum se respetan mutuamente para ser personas capaces e independientes, y son respetados como tales por las personas con las que trabajan. Los miembros del equipo de Scrum tienen el valor de hacer lo correcto y de trabajar en problemas complejos.

### Aplicación de colaboración empresarial

Al tener una mejor comunicación con los miembros del equipo mejora los procesos creativos dentro de la empresa en la que se elabora, por eso es eficaz tener un canal con el cuál compartir información interna con el equipo de trabajo.

Una de las aplicaciones para llevar una mejor comunicación es la siguiente:

**Slack**

Esta es una plataforma que permite interactuar con los miembros del equipo, así como puedes conectar las herramientas que se llega a ocupar dentro del desarrollo, así como sus servicios.

Según (Martín, 2023) Slack cuenta con las siguientes características.

* Permite tener conversaciones organizadas (enviar mensajes y compartir archivos, crear canales para equipos, oficinas, proyectos, etcétera).
* Puedes buscar mensajes en el historial de conversaciones del equipo de Slack.
* Añadir aplicaciones al espacio de trabajo para conectarlas. De esta forma, te mantendrás organizado y productivo.
* Realizar llamadas y vídeo llamadas con cualquier otro miembro del espacio de trabajo, así como compartir la pantalla si necesitas mostrarle algo.

## Propuesta de solución

Con base en la investigación anterior para la realización de la aplicación web, se requirió de la tecnología de Vue para el desarrollo del frontend, con ayuda del framework de Vue Bootstrap, utilizando TypeScript y arquitectura limpia, haciendo uso del IDE Visual Studio Code, esto con el fin de tener una mejor interacción con el usuario, utilizando a su vez el tipado estricto de TypeScript y la arquitectura limpia parque el proyecto sea sostenible en el tiempo. Así como se optó por PostgreSQL y así poder acceder a la información almacenada, además de utilizar Spring Boot como apoyo para el backend con ayuda de IntelliJ IDEA para su desarrollo.

Se determinó usar estas herramientas por lo fáciles que son, como lo es con Spring para separar los microservicios y que sea más fácil reutilizarlos. Así mismo de lo rápido que es en ejecutar y el poder conectarse a una base de datos. De igual forma con Vue, ya que se ahorra un poco más de tiempo y es mucho más fácil de usar con las directivas que tiene, esto lo que hace es extender el HTML, creando componentes, clases y atributos con funcionalidades ya dadas.

Se optó por utilizar la metodología SCRUM durante la cual se ejercerá el rol del Developer Team, se tendrá sesiones diarias en las cuales se medirá el progreso de las tareas asignas a cada desarrollador y así como reuniones mensuales con la clienta.

|  |
| --- |
| 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** |

Blandino, G. (1 de 4 de 2023). *Pixarprinting*. Obtenido de Pixarprinting: https://www.pixartprinting.es/blog/figma-que-es/

Borges, S. (26 de 09 de 2015). *infranetworking*. Obtenido de infranetworking: https://blog.infranetworking.com/5-razones-por-las-cuales-debes-usar-postgresql/

Fernández, E. C. (25 de 10 de 2022). *tokioSchool*. Obtenido de tokioSchool: https://www.tokioschool.com/noticias/spring-boot/#:~:text=Spring%20Boot%20cuenta%20con%20una,back%2Dend%20de%20forma%20independiente.

Galina, P. (21 de 05 de 2021). *iebschool*. Obtenido de iebschool: https://www.iebschool.com/blog/herramientas-prototipado-analitica-usabilidad/

Jesus. (11 de 10 de 2022). *Dongee*. Obtenido de https://www.dongee.com/tutoriales/que-es-vue-js-para-que-sirve/

JetBrains. (2024). *JetBrains*. Obtenido de JetBrains: https://www.jetbrains.com/es-es/datagrip/features/mysql.html#:~:text=DataGrip%3A%20una%20herramienta%20GUI%20para,en%20Windows%2C%20macOS%20y%20Linux.

Martín, E. (2023). *grupocibernos*. Obtenido de grupocibernos: https://www.grupocibernos.com/blog/7-aplicaciones-de-comunicacion-y-trabajo-en-equipo

Rosal, M. (31 de 05 de 2023). *ceeivalencia.emprenemjunts*. Obtenido de Centro Europero de Empresas e Innovación de Valencia: https://ceeivalencia.emprenemjunts.es/?op=8&n=25218

Schwaber, K., & Sutherland, J. (2020). *La Guía Scrum.*

Trucoteam. (2024). *trucoteca*. Obtenido de trucoteca: https://trucoteca.com/en/que-ventajas-ofrece-intellij-idea/

|  |
| --- |
| **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:

