**Segunda Evaluación Sumativa 25%**

**Informe técnico de planificación y diseño de producto software.**

**FECHA DE ENTREGA FINAL: 07-10-2024**

Instrucciones generales

1. La actividad **DEBE** ser desarrollada **SIN EXCEPCIÓN** en forma grupal (en grupos de **3** o **4** integrantes) con evaluación individual.
2. El formato del informe debe cumplir con la estructura definida en su totalidad y no se recibirán documentos incompletos.
3. Como todo informe, deberá estar redactado en tercera persona, contener portada (título, nombre de los integrantes, fecha), índice de contenidos. Eventualmente y cuando corresponda, podrá agregar glosario y bibliografía en el formato APA.
4. Deberá entregarse por plataforma en hoja tamaño carta en **Formato WORD**. La fuente para los textos no titulares es Calibri 12 o Arial de 11, interlineado de 1,5 y justificado. Los márgenes superior e inferior de 2,5 cm y el izquierdo y derecho de 3 cm.
5. El grupo deberá realizar una investigación adecuada en el que obtendrán por parte de un tercero, representante de un área de negocios o afectado por **un problema REAL que requiere automatización**, una serie de requerimientos que les permitan modelar una propuesta de solución.
6. Se deberá fundamentar y detallar las etapas, tareas y actividades, según el ciclo de vida del software.
7. Cuando corresponda, indique la(s) fuente(s) bibliográficas de acuerdo con Norma APA.

Componente 1: Idea y requerimientos

**Paso 1.1: Elección de una problemática o necesidad**

En esta actividad los estudiantes deben establecer una problemática en un mínimo de una página, con la fuente y formato indicado en las instrucciones. Recuerde que la metodología de proyecto implica que:

1. Un proyecto puede partir de un problema o situación que siempre es real.
2. El problema o situación puede ser entregado directamente por el docente.
3. El problema o situación que se pretende resolver debe ser abordado por los estudiantes como un proyecto, que debe responder a la estructura de una solución de software en sus distintas etapas.

**Paso 1.2: Análisis del contexto de mercado**

Una vez seleccionada la problemática, se debe desarrollar la documentación del contexto de mercado, utilizando:

* Análisis FODA

Se efectuará una investigación que permita establecer una situación de negocio que requiera una solución automatizada mediante software. Además, esta investigación se debe respaldar con la primera versión del levantamiento de requerimientos, utilizando al menos 3 distintas herramientas de captura de requisitos. Esto debe quedar documentado y anexarse al informe.

Deberán formular los objetivos del proyecto (generales y específicos), los cuales tienen como propósito indicar los resultados que se pretenden alcanzar.

1. Listado de Requerimientos funcionales, no funcionales, de infraestructura, de sistema y/o su equivalente en historias de usuarios
2. Lineamientos de la propuesta de solución.
3. Análisis de Factibilidad (a lo menos Técnica y Económica). **Cualquier factibilidad que no sea positiva, establecerá imposibilidad de continuar con las demás etapas de este informe y deberán cambiar el proyecto**.

Componente 2: Formulación

**Definición del problema (en un único párrafo)**

Deben redactar de manera simple, pero completa y autoexplicativa.

Es decir, al leer el párrafo debe quedar súper claro el problema sin necesidad de hacer preguntas adicionales o recurrir a algún otro material.

**Objetivo Principal del Software a Desarrollar**

Deben redactar de manera simple, pero completa y clara: ¿Qué es lo que debe hacer su software?

**Objetivos específicos** **del Software a Desarrollar**

Deben redactar de manera simple, pero completa y clara unos TRES o CUATRO objetivos específicos: ¿Qué es lo que debe hacer específicamente su software para lograr el objetivo principal?

|  |
| --- |
| ***NOTA: Asegúrense de que sus objetivos sean precisos y remítanse a señalar “el propósito” y no “el para qué”. Por ejemplo, en siguiente objetivo todo lo que está de color verde estaría sobrando:*** Desarrollar una aplicación de gestión de atenciones médicas en una clínica privada, **para lograr una atención eficiente de los pacientes de la clínica.**  ***Además, recuerden que se trata de los objetivos que tiene su software, no de sus objetivos personales como alumno. Por ejemplo, sería incorrecto señalar como objetivo:*** ”Buscar y aplicar los patrones de diseño más adecuados para la solución.” ***Ya que este sería un objetivo de ustedes, pero no es de la aplicación. Un objetivo válido de la aplicación sería:*** ”Implementar un CRUD ágil, seguro y confiable.” |

Componente 3: Modelamiento (\*)

**Diagrama de Casos de Uso** (Debe incluir diagrama y descripciones)

**Diagrama de Actividades** (Debe incluir todos los diagramas necesarios para después simplemente entregarlos a su equipo de programadores)

**Diagrama de Clases** (Debe incluir todos los diagramas necesarios para después simplemente entregarlos a su equipo de programadores)

Recuerde que su modelamiento de solución debe atender específicamente a los requerimientos planteados y no deben aparecer requerimientos imaginarios que no fueron planteados.

También debe existir consistencia entre los diagramas; es decir, El diagrama de actividades debe describir con exactitud el proceso referido en diagrama de casos de uso; a su vez, en el diagrama de clases los métodos que sean incluidos deben ser consistentes con los diagramas de actividades y los atributos deben poder satisfacer todos los requerimientos establecidos. En general, se espera que **exista consistencia absoluta entre los requerimientos y los 3 diagramas solicitados**.

|  |
| --- |
| **(\*)** Para la componente 3, ustedes pueden utilizar a discreción los patrones de diseño que consideren necesarios o adecuados, en este sentido no hay límites ni restricciones. Esta información pudiera ayudarles en este sentido; son 3 documentos que sintetizan bastante bien el uso de los patrones de Diseño:  El más sencillo: **Guía rápida de patrones de diseño**  <https://thepowerups-learning.com/guia-rapida-de-patrones-de-diseno/> Versión Intermedia y descargable en PDF: Guía de Patrones de Diseño<https://suriweb.com.ar/archivos/modelado/Objetos/Guia-de-Patrones-de-Disenio.pdf>El más completo: Guía Exhaustiva de Patrones de Diseño de JavaScript (aunque los códigos están hechos por javascript, es fácilmente adaptable a cualquier lenguaje de programación)<https://kinsta.com/es/blog/patrones-de-diseno-javascript/> |

|  |
| --- |
| PARA TENER EN CONSDIERACIÓN: ¿Y QUE VENDRÁ DESPUÉS?EVA3: PROTOTIPO FUNCIONAL **Implementación de un SW funcional:** Un prototipo funcional es un modelo interactivo y comprobable de un producto que funciona; debe incluir interfaces visuales y operativas funcionales, se debe entregar un producto instalable y que funcione en (o desde) cualquier computador.  **Implementación de la BD:** De acuerdo, al modelamiento realizado deberá implementar la Base de Datos respectiva. Ustedes determinan en qué lo hacen: SQL Server, MySQL, MongoDB, CouchDB, Elasticsearch, RavenDB, etc.  EVA4: ANÁLISIS DE ESTADO, COMPATIBILIDAD Y PLAN DE PRUEBAS |

**Rúbricas:**





|  |  |
| --- | --- |
| **Elementos de Trabajo:**    Para la confección de los **diagramas** de caso de uso y de clases puedes utilizar:  Draw.IO (<https://app.diagrams.net/>), o bien  Star UML (<https://staruml.io/>)  Los diagramas definitivos deben ser entregados como imágenes incrustadas en el documento de Word | **Bibliografía**   * Jones, C. (2011). *Estimación de costos y administración de proyectos de software*. México: McGraw Hill Interamericana. * Pressman, R. (2010). *Ingeniería del software: Un enfoque práctico.* España: McGraw Hill Interamericana. * Sommerville, I. (2011). *Ingeniería de software*. México: Pearson Educación. |