**CONSIGNA del PROYECTO FINAL**

**OBJETIVO:**

Al finalizar la actividad, el estudiante analiza, diseña y sustenta un software elaborado para el ámbito comercial, utilizando las Estructuras de Datos y Algoritmos estudiados en el curso y que solucione un problema presentado en una empresa.

El software será desarrollado en grupos de 5 integrantes y será presentado en un **Portafolio** en dos partes:

* Parte I: Informe calificado de manera grupal
* Parte II: Sustentación oral

***Parte I: Informe***

* El informe debe ser presentado en formato Word y tener presente los siguientes puntos:

1. Aspectos generales

2. Requerimientos de la aplicación

3. Diseño de la aplicación

4. Conclusiones y Recomendaciones

* Formato: archivo Word con tipo de letra Calibri en tamaño 11, con espaciado simple y siguiendo la estructura estipulada en la plantilla del Informe del Proyecto Final.

***Parte II: Sustentación oral***

* La sustentación oral se realizará apoyándose en 2 tipos de archivos, los cuales serán presentados en la clase presencial (**semana 18**):
  + PPT: debe considerar un resumen del informe.
  + Software: prototipo desarrollado bajo lenguaje de programación JAVA.

**2. Presentación**

1. **Formato:** Archivo Word, PPT y software (Proyecto Java).
2. **Canal:** Entrega virtual vía plataforma UTP+Class y exposición en clase presencial.
3. **Fecha a presentar:** 
   * + La archivos relacionados a la entrega se subirán a UTP+Class.
4. **El primer avance del Portafolio:** Semana 16
   * + Incluye el Informe Capítulo 1 y 2
     + Primer Avance del Prototipo
   1. **La entrega del Trabajo Final:** Semana 18
      * Incluye Informe Completo
      * Aplicación terminada que incluye
        1. Creación del Formulario Principal y menú principal
        2. Creación de la jerarquía de clases.
        3. Diseño de todos los formularios con sus controles.
        4. Implementación de Clases para importación y exportación de datos (archivos planos)
        5. Formulario de Registro para su tema
        6. Formulario de Búsqueda para su tema
        7. Formulario de Reportes para su tema
5. **Parámetros para la exposición:** Considerar lo siguiente:

* Tiempo de exposición: entre 8 a 10 minutos por grupo.
* La presentación debe incluir un resumen del informe.
* Debe demostrar funcionamiento y conocimiento del software desarrollado.

**CONSIDERACIONES TÉCNICAS**

1. El programa debe estar creado con las librerías de Java Swing *(formularios y controles visuales).*
2. La información el programa consulta, actualiza o grabar, se encuentra en archivos de texto (CSV o TXT). No accede a ninguna base de datos, tampoco se utilizará archivos XML.
3. Los formularios de mantenimiento utilizarán las Librerías de Java para poder acceder a los archivos planos (csv, texto.) y realizar las operaciones de básicas de búsquedas, inserción.
4. Los formularios de mantenimiento o consultas utilizarán estructuras de datos dinámicas vistas en el curso:
   1. Listas enlazadas circulares dobles, colas, pilas, árboles ABB, árboles AVL, para mostrar y filtrar la información que el usuario solicita.
5. El programa estará estructurado en Paquetes y Clases.

**ANEXO 01**

**CASO DE ESTUDIO**

**ESTADISTICAS - MUNICIPALIDADES**

La tienda UTP contrato los servicios de su equipo de trabajo para diseñar y construir el nuevo sistema de información para ingresar datos estadísticos de las municipalidades del Perú.

La aplicación estará construida con **Formularios** **de** **Java Swing** y utilizará archivos de texto como orígenes de datos.

**DESCRIPCIÓN DE ENTIDAD DEL GOBIERNO**

Se han realizado encuestas a todas las municipalidades del Perú para recabar datos con relación a las siguientes categorías:

* Definición de variables globales (4 campos)
* Datos generales de la municipalidad de centro poblado (2 campos)
* Personal y servicios municipales (2 campos)
* Transferencia de recursos y gastos ejecutados (2 campos)
* Principal vía de acceso y ámbito geográfico (2 campos)

El gobierno ha sido informado que han municipalidades para las que no se han recabado los datos de las encuesta.

El sistema informático debe permitir

* + - Ingresar los datos para las municipalidades faltantes
    - Buscar registro(s) utilizando párametros
    - Generar los siguientes reportes de los datos más actuales.
* Cantidad de municipalidades por provincia
* Cantidad de municipalidades por departamento
* Cantidad de municipalidades por medio de transporte
* Utilidad total (ingreso – gasto) por municipalidad

**FUNCIONAMIENTO DEL PROGRAMA**

El **Encargado y Asistente** del área de estadísticas necesita que el programa le permita:

1. Importar los archivos planos hacia la aplicación.
2. Contar con un **módulo** para agregar datos de municipalidades faltantes
3. Contar con un **módulo** para realizar búsquedas por diferentes parámetros.
4. Contar con un **módulo** para generar reportes por diferentes parámetros.
5. Exportar los resultados de las búsqueda y/o reportes realizadas a archivos **pdf** y **csv**.

**La aplicación tendrá los siguientes formularios:**

|  |
| --- |
| **Formulario Principal**  El formulario principal posee un **menú principal** y permite el ingreso de los usuarios y es el punto desde el cual los usuarios acceden a los demás formularios a través de un menú principal. |

|  |
| --- |
| **Formulario Registro de datos Municipalidades Faltantes**  Permite registrar datos   * + - El administrador registra y guarda los datos de nuevas municipalidades |
| **Nota:**  - El programa **guarda** los datos de las municipalidades en el archivos **muni\_actualizado.txt**  - Además muestra, los datos de los municipalidades registradas.  - El archivo **muni\_actualizado.txt** contiene los datos de las municipalidades que ya existían y los de la recientemente agregadas. |

|  |
| --- |
| **Formulario de Búsqueda de las Municipalidades**  Formulario de Consulta donde el Administrador consulta información de las municipalidades.  Las consultas se pueden filtrar por los parámetros:   1. Año 2. Departamento 3. Provincia 4. Cobertura de Radio 5. Medio de Transporte   El administrador puede ingresar alguno de los 5 parámetros, o combinaciones de ellos y al presionar el botón **Buscar,** el programa mostrará la información de las municipalidades que coincidan con los parámetros de búsqueda. |
| **Funcionamiento:**  - El programa **importa** los datos de las municipalidades desde el archivo.  - Los datos son importados en **estructuras dinámicas de datos de objetos** (listas enlazadas circulares, árboles ABB, árboles AVL) que se cargan en la memoria de la computadora.  - El programa mostrará por pantalla las **colecciones de objetos** que coincidan con los parámetros de búsqueda ingresados por el administrador.  **Nota:**  - **No se utilizará** ninguna de las colecciones de Java: ArrayList, List, Map, Set, etc.  - Los formularios no podrán leer o escribir los datos directamente de los archivos, los datos deben ser cargados/importados primero en estructuras de datos.  - Todas las estructuras de datos, deben estar definidas en TADs y deben ser estructuras genéricas (clases genéricas).  - Sólo se pueden usar archivos planos (txt, csv) **no** se puede utilizar otro tipo de archivos (XML, etc.) ni se puede usar base de datos.  - Investigar sobre los métodos de ordenamiento avanzados *(quicksort, mergesort, etc),* e implementarlo en las estructuras de datos dinámicas. |

|  |
| --- |
| **Formulario de Reportes de los Municipalidades**  Formulario de Reportes el Administrador realiza cálculos con la información de los municipalidades.  Los reportes que se pueden generar con los datos más actuales son:   1. Cantidad de municipalidades por provincia 2. Cantidad de municipalidades por departamento 3. Cantidad de municipalidades por medio de transporte 4. Utilidad total (ingreso – gasto) por municipalidad   Todos los reportes de pueden **Filtrar**, el administrador puede ingresar alguno de los 4 parámetros, o combinaciones de ellos y al presionar el botón **Generar,** el programa mostrará la información de los cálculos que coincidan con el tipo de reporte y parámetros ingresados. |
| **Funcionamiento:**  - El programa **importa** los datos de las municipalidades desde el archivo.  - Los datos son importados en **estructuras dinámicas de datos de objetos** *(listas enlazadas circulares dobles, árboles ABB, árboles AVL.)* que se cargan en la memoria de la computadora.  - El programa mostrará por pantalla las **colecciones de objetos** que coincidan con los parámetros de búsqueda ingresados por el administrador.  **Nota:**  - **No se utilizará** ninguna de las colecciones de Java: ArrayList, List, Map, Set, etc.  - Los formularios no podrán leer o escribir los datos directamente de los archivos, los datos deben ser cargados/importados primero en estructuras de datos.  - Todas las estructuras de datos, deben estar definidas en TADs y deben ser estructuras genéricas (clases genéricas).  - Sólo se pueden usar archivos planos (txt, csv) **no** se puede utilizar otro tipo de archivos (XML, etc.) ni se puede usar base de datos.  - Todos los reportes se pueden exportar hacia archivos planos (txt, csv). |