

JUEVES 9 DE NOVIEMBRE DEL 2019

Luciano Lopez Duran, Edwar Zevallos Rojas

UPC: Universidad peruana de ciencias aplicadas

FILEEX

INFORME DE TRABAJO FINAL

Contenido

[Introducción 2](#_Toc24189388)

[Objetivo del estudiante 2](#_Toc24189389)

[Capítulo 1: Presentación 3](#_Toc24189390)

[Capítulo 2: Marco conceptual 3](#_Toc24189391)

[Capítulo 3: Diseño del proyecto 3](#_Toc24189392)

[ Diseño de un plan de proyecto de desarrollo 3](#_Toc24189393)

[Capítulo 4: Diseño del producto 4](#_Toc24189394)

[ Definición de las necesidades 4](#_Toc24189395)

[ Definición de requisitos funcionales y no funcionales 4](#_Toc24189396)

[o Requisitos Funcionales 4](#_Toc24189397)

[o Requisitos No funcionales 4](#_Toc24189398)

[ Elaboración de diagrama de clases 5](#_Toc24189399)

[ Diseño de la interfaz de usuario 7](#_Toc24189400)

[ Diseño de tipos de datos abstractos 8](#_Toc24189401)

[ Selección de estructuras de datos 8](#_Toc24189402)

[Capítulo 5: Diseño del proceso de desarrollo de la solución 8](#_Toc24189403)

[ Diseño de la arquitectura del software 8](#_Toc24189404)

[ Implementación de las funcionalidades del software. 8](#_Toc24189405)

[ Pruebas del software 8](#_Toc24189406)

[Conclusiones 12](#_Toc24189407)

[Bibliografía 12](#_Toc24189408)

[Anexos 12](#_Toc24189409)

## Introducción

Durante la presentación del proyector se definida los objetivos del alumno con respecto al proyecto realizado con la finalidad de mostrar la utilidad de lo aprendido durante el curso de Algoritmo de Estructura y Datos.

Se realizará una presentación sobre el proyector en el cual se mostrará las herramientas usada para poder cumplir el requerido en el curso.

## Objetivo del estudiante

Los objetivos del curso que debe tener en cuenta los alumnos son:

* La implementación de nodos.
* El uso adecuado de témplates.
* La manera correcta de codificar los árboles, ya sea en un ARBOLAVL, Binary Heap, Binary Max.
* Investigación por parte del alumno para determinar poder hacer uso de las librerias como System::IO, ya que hicimos usar de Directory, File y Path para determinar los nombre, año, día, mes y tamaño de un archivo.

## Capítulo 1: Presentación

Durante este proyecto se presentada las herramientas utilizadas para poder hacer un uso correcto del ArbolAVL, con la finalidad de que nos ayude a almacenar los diversos datos. Este proyecto se basa en una exploración de archivos, la cual tiene como finalidad almacenar cada dato en un nodo de un ArbolAVL. Por consiguiente, el alumno podrá aprender hacer un uso correcto de implementación al momento de insertar, sin necesidad de hacer uso del GET en el trabajo realizado, ya que al almacenar otro tipo de variable que no tenga esta función, el proyecto podría fallar.

## Capítulo 2: Marco conceptual

Luego, se realizará un diseño de solución con las asignaciones de recursos, milestones y issues con la finalidad de realizar un plan de solución en la ingeniería.

Posteriormente, se hará un diseño de producto con las definiciones en el impacto de salud pública, seguridad, bienestar, cultura, sociales, ambientales y económicos. Además, se definida los requisitos funcionales y no funciones. También, mostrara el uso de diagramas de clases usados en el proyector, la interfaz de usuario, diseño de datos abstractos y selección de estructura de datos.

Luego se mostrará el diseño de arquitectura del software, implementación del software y la prueba de compilación sobre el proyecto realizado.

Finalmente, se llegará a una conclusión en el cual el alumno dará las opiniones sobre la utilidad del trabajo y como este le ayuda al estudiante a poder dominar las diversas maneras de lo estudiado durante el ciclo.

## 

## Capítulo 3: Diseño del proyecto

### Diseño de un plan de proyecto de desarrollo

**Hito: En este hito se tuvo pensado crear un plan para poder cumplir lo requerido en el corto plazo que nos queda en el curso:**

El trabajo tiene como finalidad hacer la creación de un explorador de archivos. Con la finalidad, de hacer uso de lo estudiado en la clase de Estructura de Algoritmo y Datos los cuales son el uso de nodos y árboles, ya sea binario heap o binario search. Se tiene planeado hacer una repartición para cumplir con el trabajo asignado. La cual está estructurada de la siguiente manera, con el cual tenemos una idea a programar y con el tiempo establecido:

## Capítulo 4: Diseño del producto

### Definición de las necesidades

Tras los múltiples intentos de reformas, no cabe duda que existe una deuda que se tiene con la sociedad en materia de salud pública. Se requiere un cambio estructural en el que el estado aborda la salud y los medios con los que los brinda. Hasta ahora solo se han logrado un conjunto de esfuerzos aislados que no logran una mejora tangible.

La complejidad y dificultad de la tarea es muy alta. Para ello se debe dividir en tareas de menor complejidad que tengan logros tangibles y medibles.

Unas de las primeras tareas a solucionar en la calidad de la atención al público. Por ello desde el curso Algoritmos y Estructuras de datos proponemos el uso de FILEEXP como software para la atención al público e los establecimientos de salud pública. Con el fin de buscar archivos, historiales e información sobre los pacientes, médicos, y servicios de una forma rápida y eficiente.

### Definición de requisitos funcionales y no funcionales

### Requisitos Funcionales

* + - El software debe mostrar los archivos de la carpeta seleccionada.
    - Los archivos deben mostrarse en una lista.
    - El software debe contar con un botón de selección de carpeta y otro de busqueda.
    - El software debe catalogar los archivos por Nombre, Tamaño, Extension e Iniciales del nombre del archivo.

### Requisitos No funcionales

* + - El software debe ser capaz de procesar y responder la búsqueda del operador en 3 segundo o menos, esto se medirá por medio de herramientas de Software Testing
    - El software debe ser capaz de funcionar en equipos de pocas capacidades o computadoras de gama baja.
    - El software debe ser compatible con los sistemas operativos Windows XP y superiores
    - El software debe desarrollarse aplicando patrones de diseño.
    - No Funcionales
    - El sosftware debe ser capaz de procesar y responder la busqueda del pperador en 3 segundo o menos, esto se medira por medio de herramientas de Software Testing
    - El software debe ser capaz de funcionar en equipos de pocas capacidades o computadoras de gama baja.
    - El software debe ser compatible con los sistemas operativos windows XP y superiores
    - El software debe desarrollarse aplicando patrones de diseño
    - el tiempo de aprendizaje del operador debe ser menor a 4 horas.
    - Debe contar con manual de uso breve y estructurado o un módulo de ayuda.
    - El software debe tener un diseño responsiva con el fi de garantizar la adecuada visualización en múltiples computadoras.

### Elaboración de diagrama de clases

**Almacenado:**

|  |
| --- |
| En el main se llama: |
| Almacenado almacen |
|  |
| class Almacenado |
| string Nombre; string ext; int Anio; int Mes; int Dia; long long Tamanio; char PrimeraLetra; char UltimaLetra; int Contiene; |
| SET\_Nombre(string); SET\_Ext(string); SET\_Anio(int); SET\_Mes(int); SET\_Dia(int); SET\_Tamanio(long long); SET\_PrimeraLetra(string); SET\_UltimaLetra(string); SET\_Contiene(int);  GET\_Nombre(); GET\_Ext(); GET\_Anio(); GET\_Mes(); GET\_Dia(); GET\_Tamanio(); GET\_PrimeraLetra(); GET\_UltimaLetra(); GET\_Contiene(); |

**Nodo:**

|  |
| --- |
| template<class Alma,class Int> |
| class cNodo |
| cNodo<Alma,Int>\*derecha; cNodo<Alma,Int>\*izquierda; int altura; T elemento; Alma elementos; |
| cNodo(); |

**ArbolAVL:**

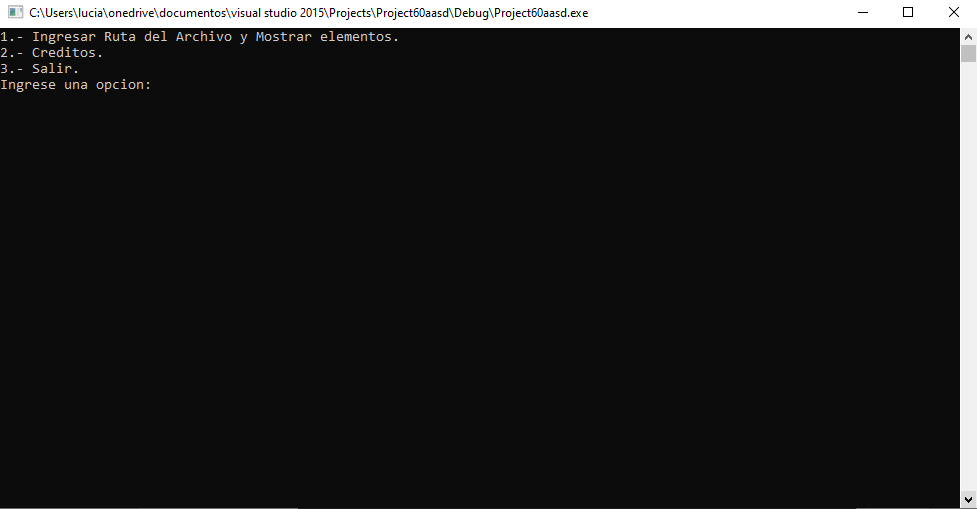
|  |
| --- |
| En el main se llama: |
| ArbolAVL<Almacenado,int>\*arbol(Imprime); |
| template<class Alma,class Int> |
| class ArbolAVL |
| cNodo<Alma,Int>\*raiz; void (\*procesar)(T); void \_insertar(cNodo<Alma,Int>\*&nodo,Alma elementos,Int elemento); int \_altura(cNodo<Alma,Int>\*&nodo); void RestauraNivel(cNodo<Alma,Int>\*&Centro,cNodo<Alma,Int>\*&Iz,cNodo<Alma,Int>\*&Dere); rotaIz(cNodo<Alma,Int>\*a,cNodo<Alma,Int>\*&b,cNodo<Alma,Int>\*&c); rotaDere(cNodo<Alma,Int>\*a,cNodo<Alma,Int>\*&b,cNodo<Alma,Int>\*&c); int \_max(int a,int b); void inOrder(cNodo<Alma,Int>\*&nodo); |
| ArbolAVL(void(\*pf)(Alma)); void Insertar(Alma elementos,Int elemento); void ImprimeOrden(); |

### Diseño de la interfaz de usuario

El proyecto esta trabajado en una consola CLR por lo tanto nuestra interfaz es la siguiente:







### Diseño de tipos de datos abstractos

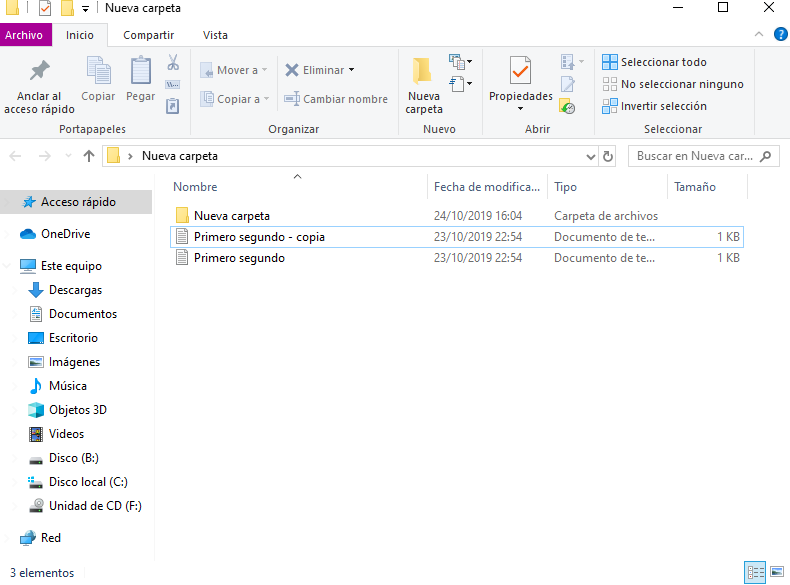
### Selección de estructuras de datos

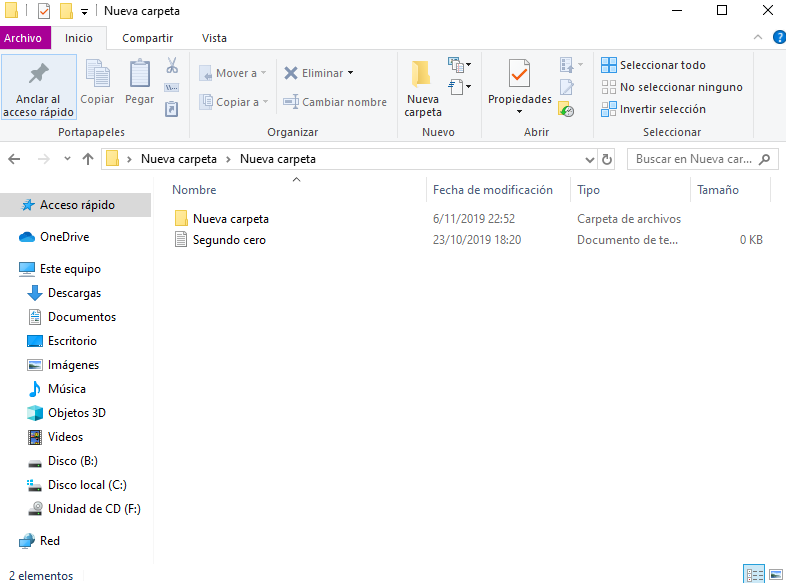
## Capítulo 5: Diseño del proceso de desarrollo de la solución

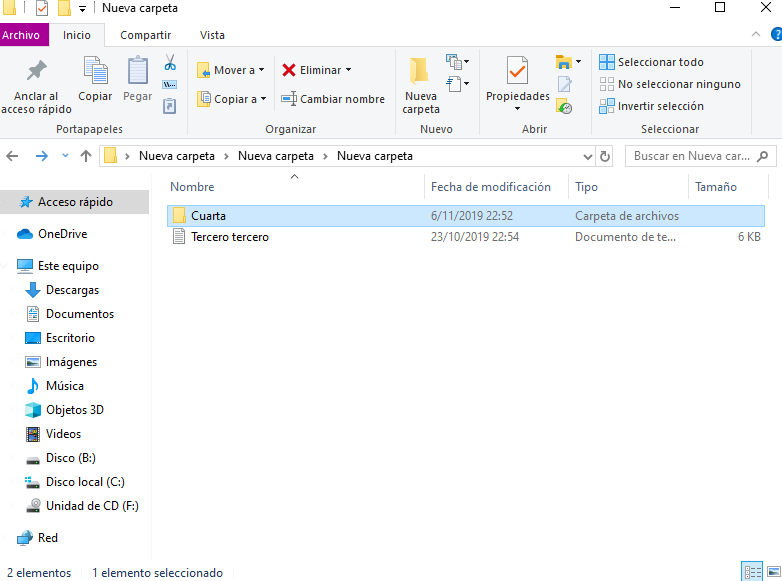
### Diseño de la arquitectura del software

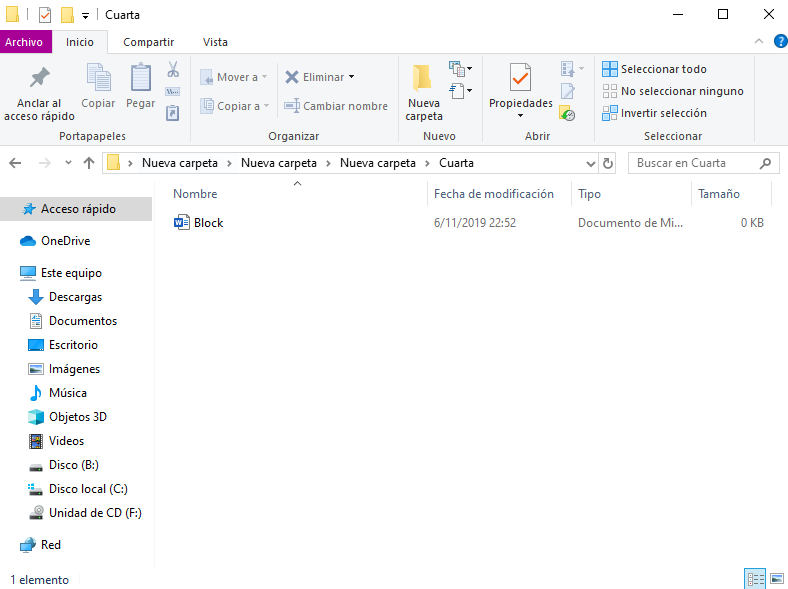
### Implementación de las funcionalidades del software.

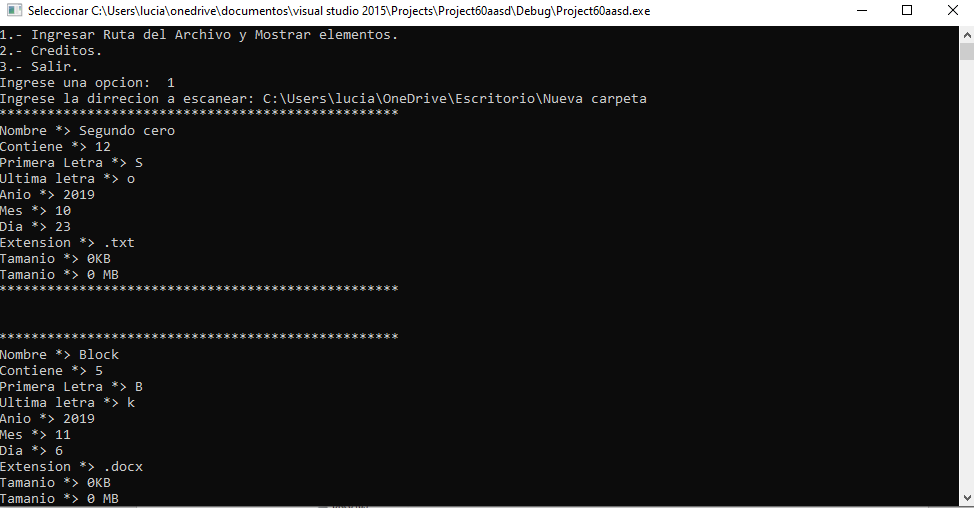
### Pruebas del software

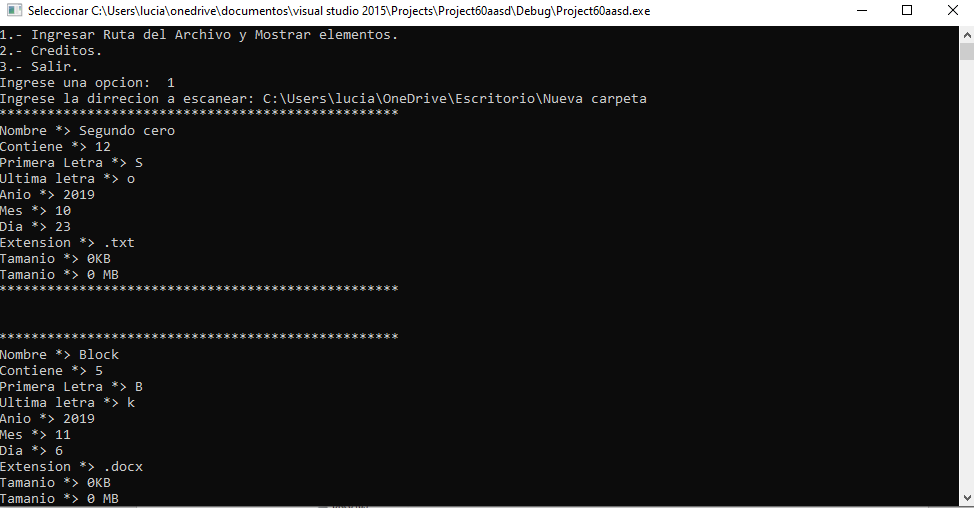


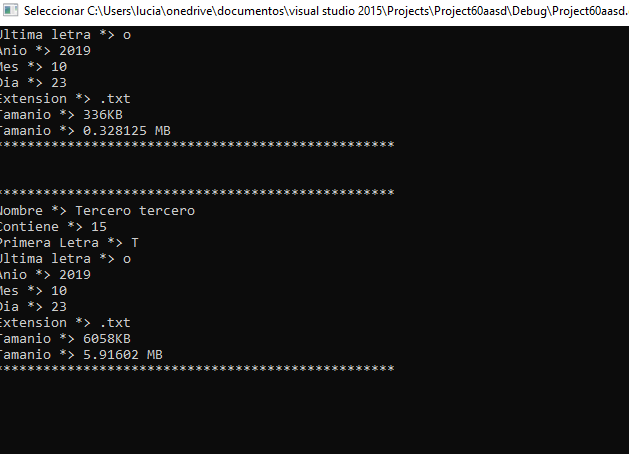












## Conclusiones

## Bibliografía

* Gabriel del Castillo Mory. Siguiente reforma: La impostergable reingenieria del sistema de salud pública en el Perú. Pacifico Business School. <http://pbs.edu.pe/noticias/siguiente-reforma-impostergable-reingenieria-sistema-salud-publica-peru-gabriel-castillo-mory/>
* Programa de prioridades de la Salud. Instituro Nacional de Estadistica e Informatica (INEI). <https://proyectos.inei.gob.pe/endes/endes2007/1.%20Introducci%C3%B3n/1.7%20Per%C3%BA%20Programa%20y%20Prioridades%20de%20Salud.html>
* Requerimientos no funcionales, PMOinformatica. http://www.pmoinformatica.com/2015/05/requerimientos

## Anexos

* Projecto Gihub: https://github.com/users/EdwarZevallos/projects/1
* Repositorio GitHub: https://github.com/EdwarZevallos/AlgoFinal