



Documento de Análisis y Diseño

Nivel 15: Cupisearch

[Felipe Otálora]

[Sebastián Florez]

Estructuras de Datos

Universidad de Los Andes

[III]



Documento de Análisis y Diseño

CupiSearch

| Versión del documento | Fecha de modificación | Responsables |
|------------------------------|------------------------------|------------------------------------|
| v1 | 18 de Marzo - 2014 | Felipe Otálora Sebastián Florez |
| | | |
| | | |



Contenido

| | |
|---|----|
| Requerimientos Funcionales | 3 |
| Análisis del modelo del mundo | 5 |
| Diseño de Interfaces | 6 |
| Interfaces del mundo de la aplicación | 6 |
| Interfaces de las estructuras de datos | 6 |
| Diseño de estructuras de datos | 7 |
| Justificación de decisiones de diseño de las estructuras de datos | 8 |
| Diseño final de la aplicación | 9 |
| Justificación de decisiones de diseño del mundo | 10 |
| Análisis de complejidad de operaciones del mundo | 12 |



1. *Requerimientos Funcionales*

A continuación se describen los requerimientos funcionales de la aplicación. La definición de estos es independiente de la tecnología que será usada para implementarlos.

| | |
|-------------------|--|
| ID | <identificador único, generalmente un consecutivo> |
| Nombre | <nombre, comienza con verbo en infinitivo> |
| Resumen | <breve descripción del requerimiento> |
| Entradas | <nombre entrada> |
| Resultados | <nombre resultado> |

| | |
|-------------------|---|
| ID | R1 |
| Nombre | Definir un conjunto de sitios fuente |
| Resumen | La aplicación le debe permitir al usuario especificar un conjunto de sitios de exploración. |
| Entradas | - El/los sitio/s fuente/s que se van a agregar |
| Resultados | Se ha agregado un nuevo sitio fuente a la aplicación |

| | |
|-------------------|--|
| ID | R2 |
| Nombre | Hacer una exploración con el Web-Crawler a partir de un conjunto de sitios fuente. |
| Resumen | La aplicación debe poder realizar una exploración dentro del conjunto de sitios web dependiendo de la profundidad de la exploración. Los resultados de la exploración deben completar el índice de búsqueda. |
| Entradas | - Profundidad: La profundidad de la exploración - Conjunto de sitios fuente: Los sitios fuente |
| Resultados | Se muestran los resultados de la búsqueda y deben completar el índice de búsqueda. |

| | |
|-------------------|--|
| ID | R3 |
| Nombre | Mostrar estadísticas de un resultado particular |
| Resumen | Se deben incluir el número de recursos explorados, tiempo total de exploración y número de búsquedas con resultados. |
| Entradas | - La exploración con los resultados |
| Resultados | Se han mostrado los resultados de la exploración dada. |

| | |
|-----------|----|
| ID | R4 |
|-----------|----|



| | |
|-------------------|---|
| Nombre | Consultar el historial de operaciones realizadas |
| Resumen | Se deben conocer el historial de las búsquedas y exploraciones realizadas: (sitios fuente, hora, fecha, duración) |
| Entradas | |
| Resultados | Se ha mostrado la información solicitada sobre el historial |

| | |
|-------------------|--|
| ID | R5 |
| Nombre | Realizar una búsqueda dado un conjunto de criterios |
| Resumen | La aplicación le debe permitir al usuario realizar una búsqueda al especificar un conjunto de sitios de exploración y criterios. |
| Entradas | - El conjunto de criterios para la exploración |
| Resultados | Se ha mostrado la información de los resultados de la búsqueda. (Se pueden seleccionar los resultados) |

| | |
|-------------------|---|
| ID | R6 |
| Nombre | Crear o eliminar una categoría |
| Resumen | La aplicación le debe permitir al usuario especificar una categoría para poderla agregar a eliminar |
| Entradas | - La categoría que se quiere agregar/eliminar |
| Resultados | Se ha agregado/eliminado una nueva categoría a la aplicación |

| | |
|-------------------|--|
| ID | R7 |
| Nombre | Adicionar o eliminar un recurso a una categoría de la lista |
| Resumen | La aplicación le debe permitir al usuario agregar o eliminar un recurso de una categoría |
| Entradas | - El recurso que se quiere agregar/eliminar - La categoría para agregar/eliminar el recurso |
| Resultados | Los cambios realizados han persistido |

| | |
|-------------------|---|
| ID | R8 |
| Nombre | Comprimir las categorías utilizando el algoritmo de Hoffman |
| Resumen | La aplicación le debe permitir al usuario comprimir las categorías utilizando el algoritmo de Hoffman |
| Entradas | |
| Resultados | Los cambios realizados han persistido |

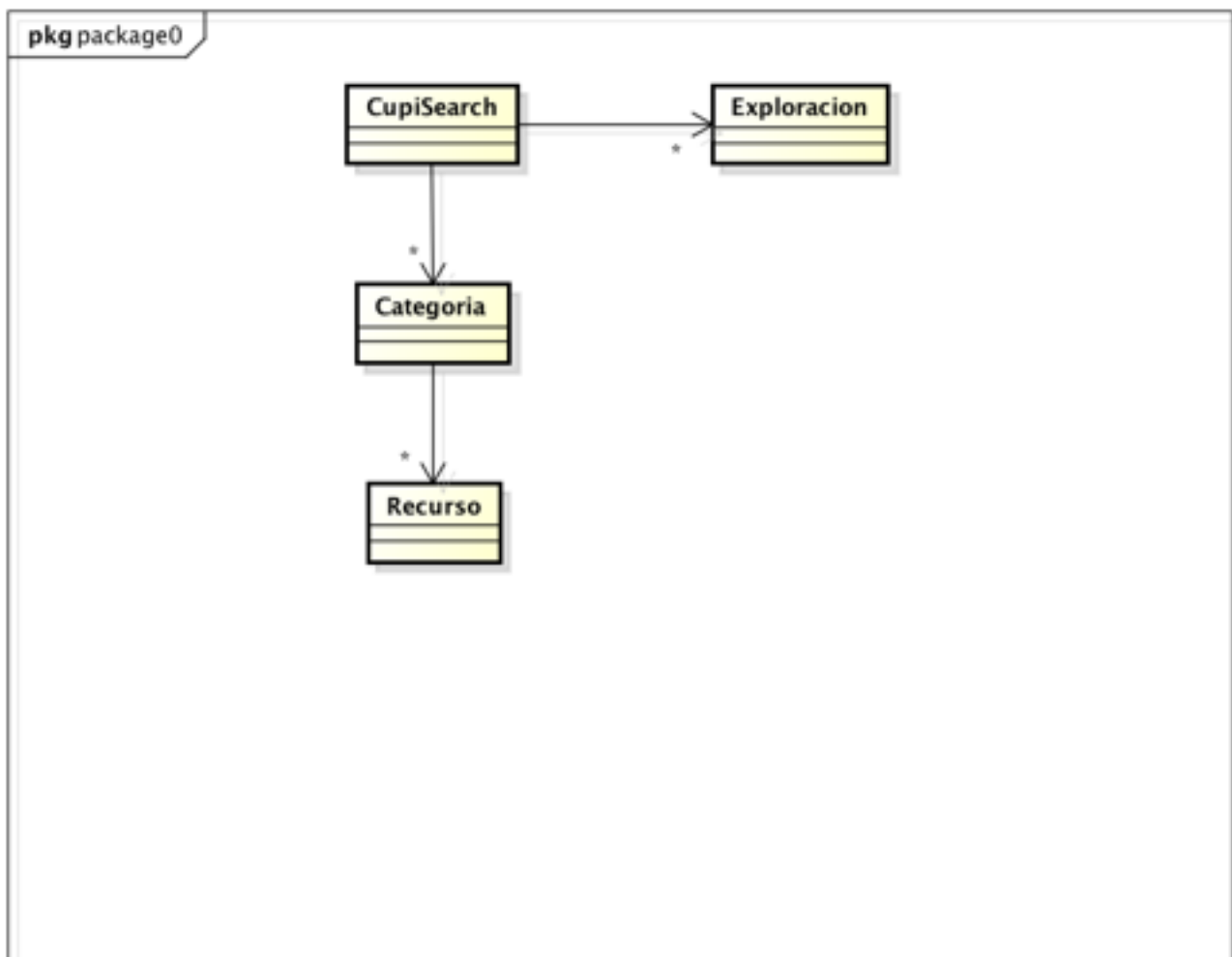


| | |
|-------------------|--|
| ID | R9 |
| Nombre | Recuperar un conjunto de categorías del servidor de persistencia |
| Resumen | La aplicación le debe permitir al usuario recuperar las categorías comprimidas desde el servidor de persistencia |
| Entradas | |
| Resultados | Se ha recuperado la información desde el servidor y los cambios se ven en la interfaz. |

| | |
|-------------------|---|
| ID | R10 |
| Nombre | Visualizar un recurso resultado de una búsqueda |
| Resumen | La aplicación le debe permitir al usuario visualizar el recurso resultado de una búsqueda |
| Entradas | - El recurso que se quiere visualizar de la búsqueda |
| Resultados | Se ha visualizado la información del resultado |

2. Análisis del modelo del mundo

Diagrama de clases (UML) de la abstracción del problema que se pretende solucionar. Solo se tienen en cuenta entidades, características y relaciones relevantes para el mismo. Aún no se toman decisiones de diseño.

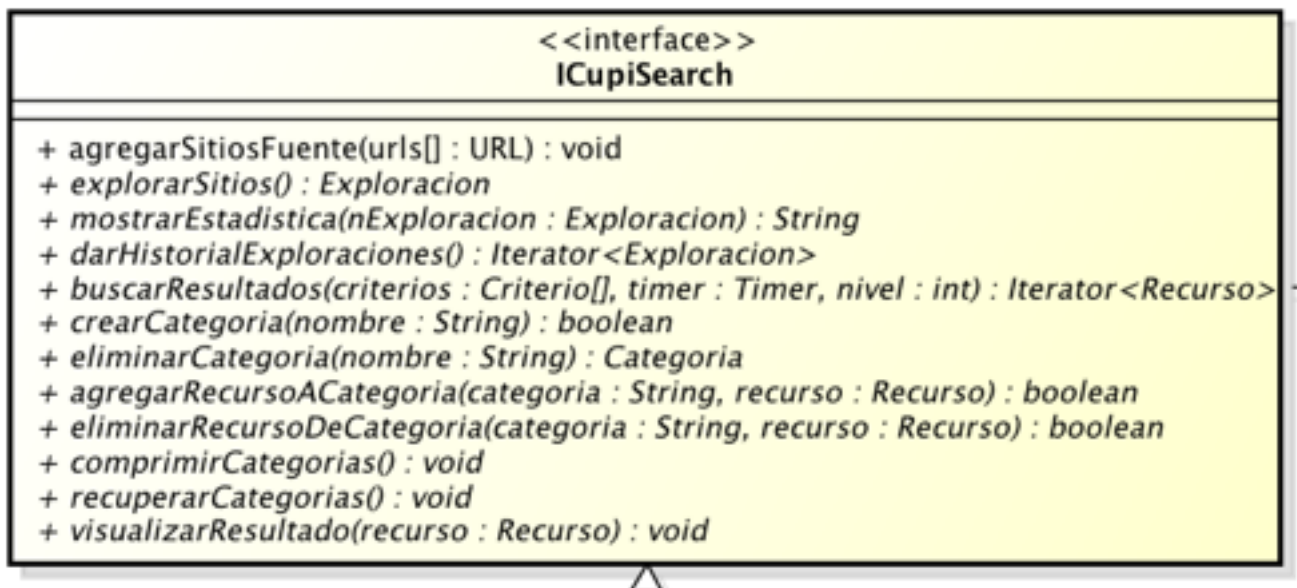




3. Diseño de Interfaces

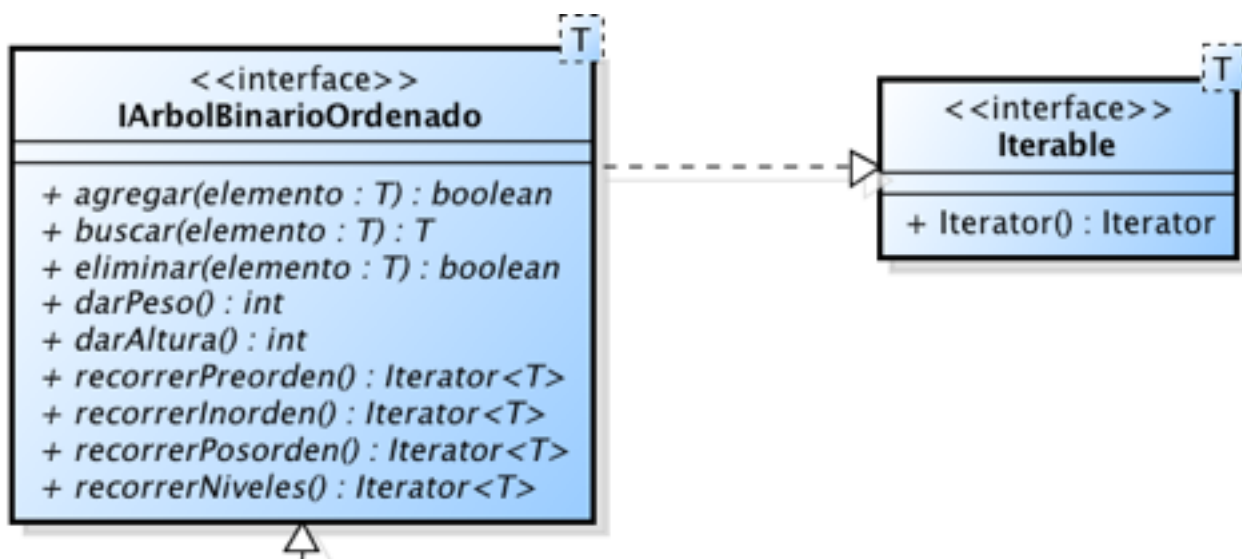
1. Interfaces del mundo de la aplicación

A partir de los requerimientos funcionales, defina las operaciones (a través de métodos) que debe proveer el mundo del problema. Incluya parámetros, tipos de retorno y documentación completa de cada operación (incluyendo precondiciones).



2. Interfaces de las estructuras de datos

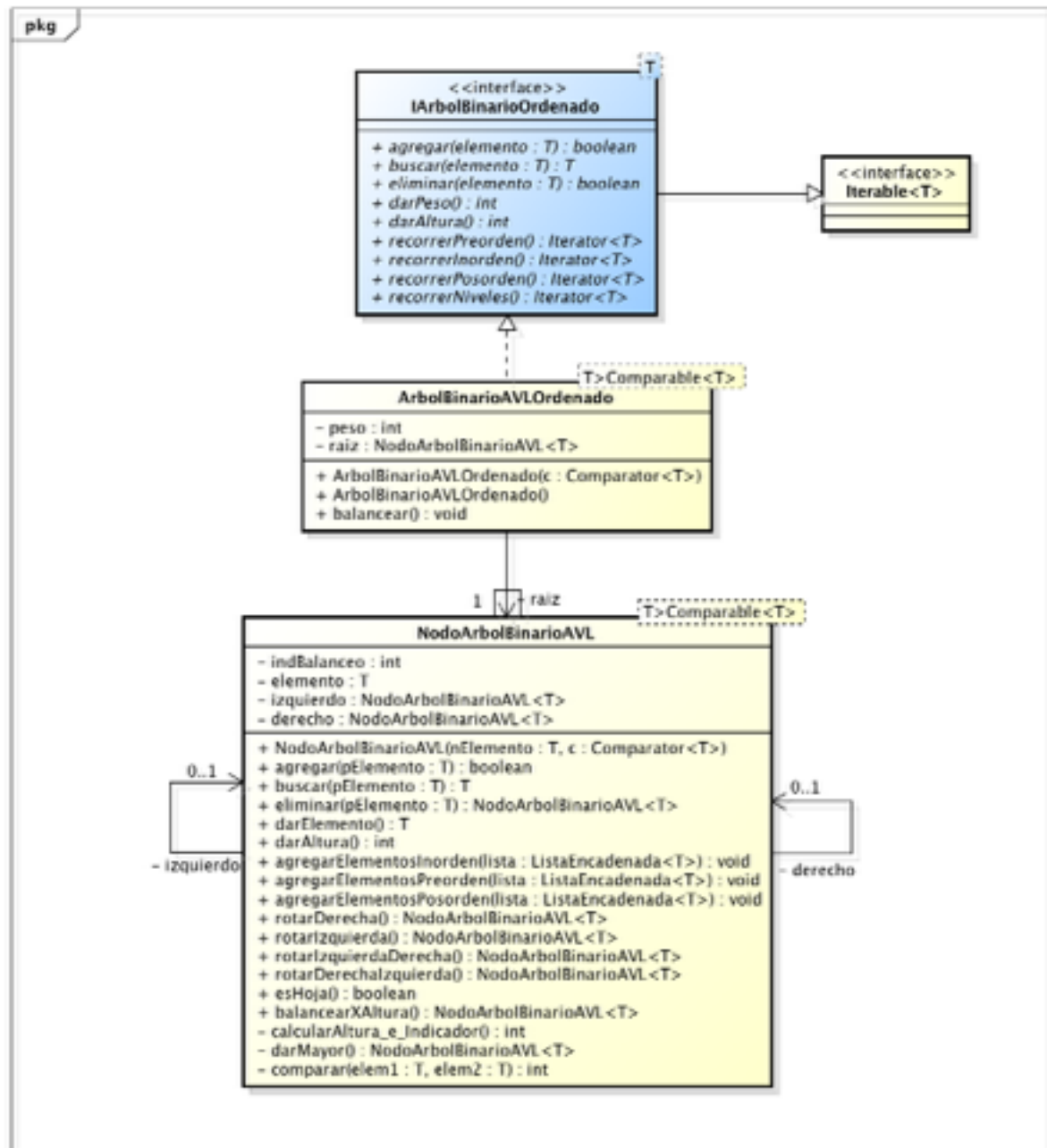
Seleccione cuáles son las estructuras de datos genéricas que debe utilizar y las operaciones que estas deben proveer para minimizar la complejidad temporal de las operaciones definidas en el numeral 3.1.





4. Diseño de estructuras de datos

Describe el detalle de las Estructuras de Datos que va a utilizar en su solución, a través de un diagrama UML de las mismas. No olvide incluir invariantes si las requiere.





5. *Justificación de decisiones de diseño de las estructuras de datos*

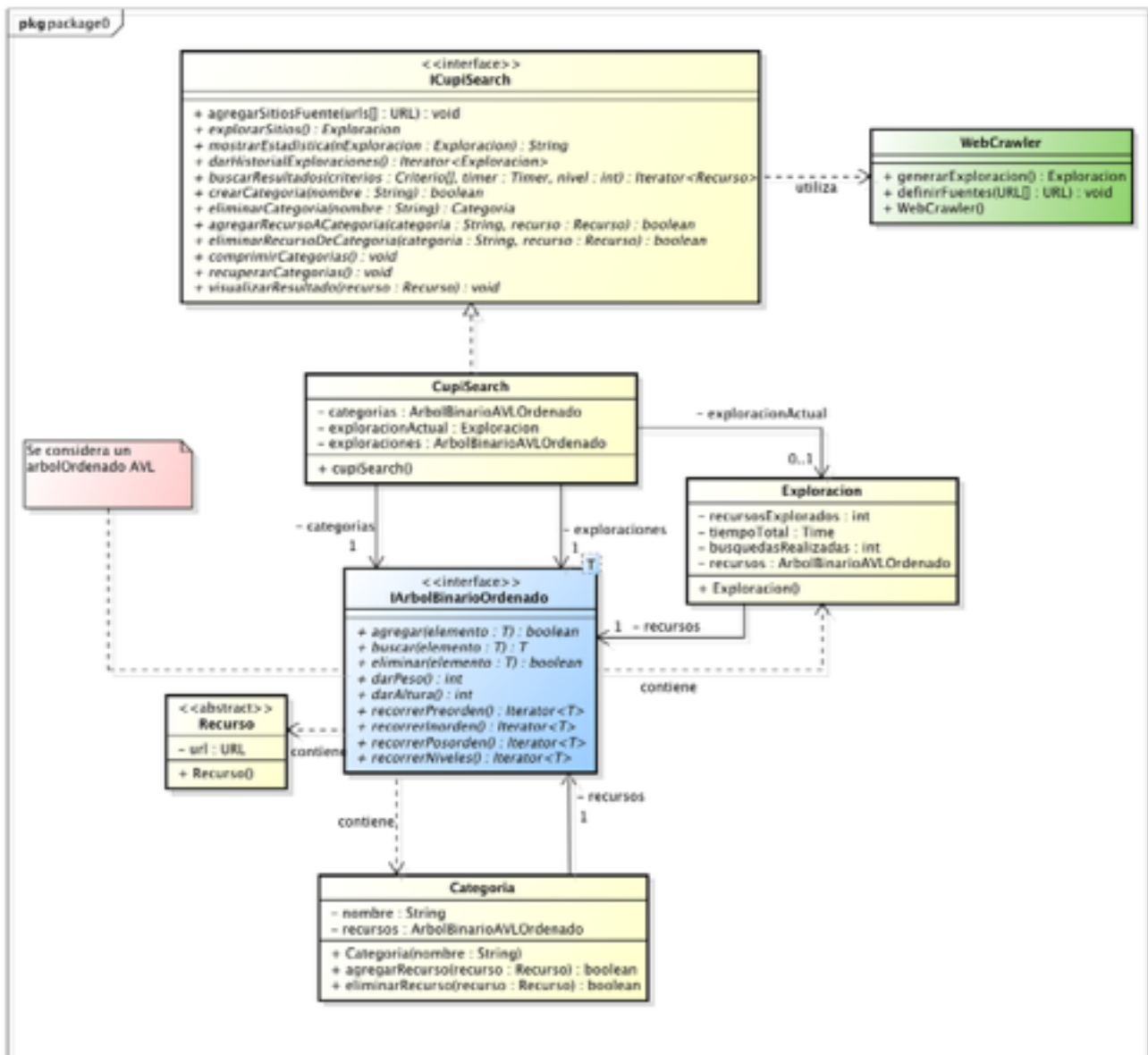
Describe en detalle los aspectos que tuvo en cuenta para seleccionar y diseñar las estructuras de datos descritas en el punto anterior.

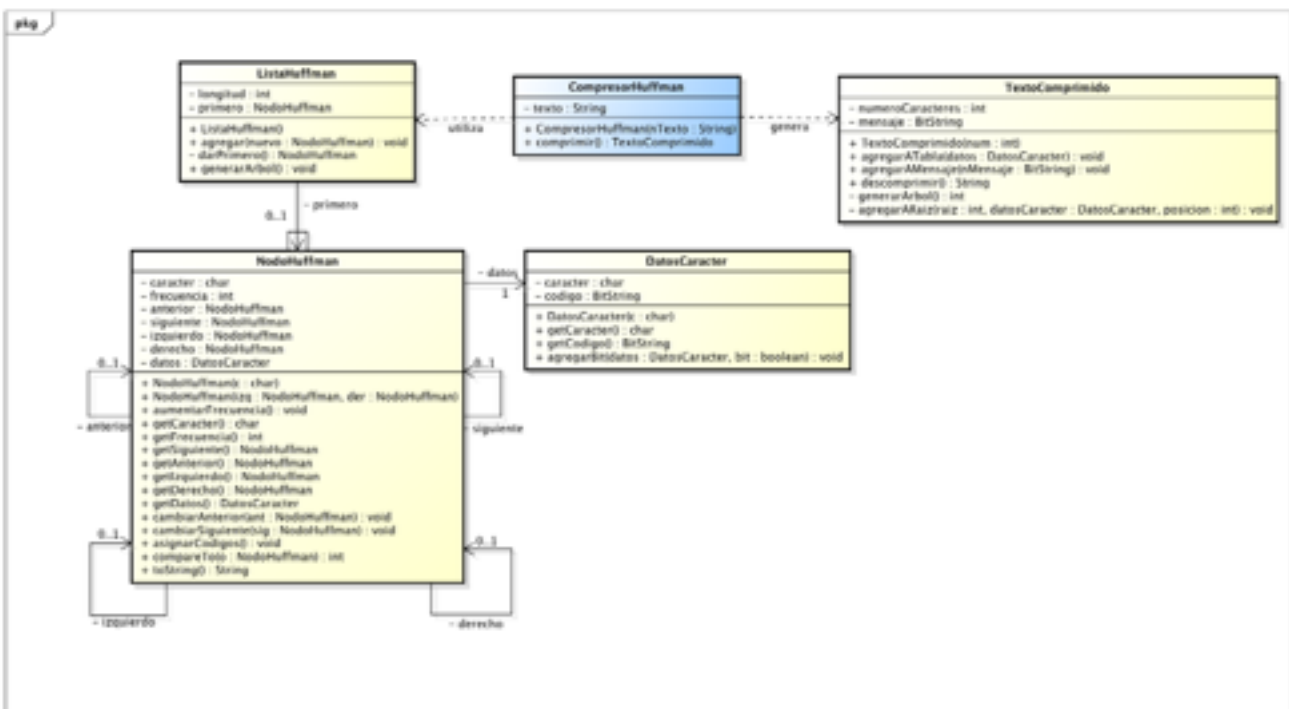
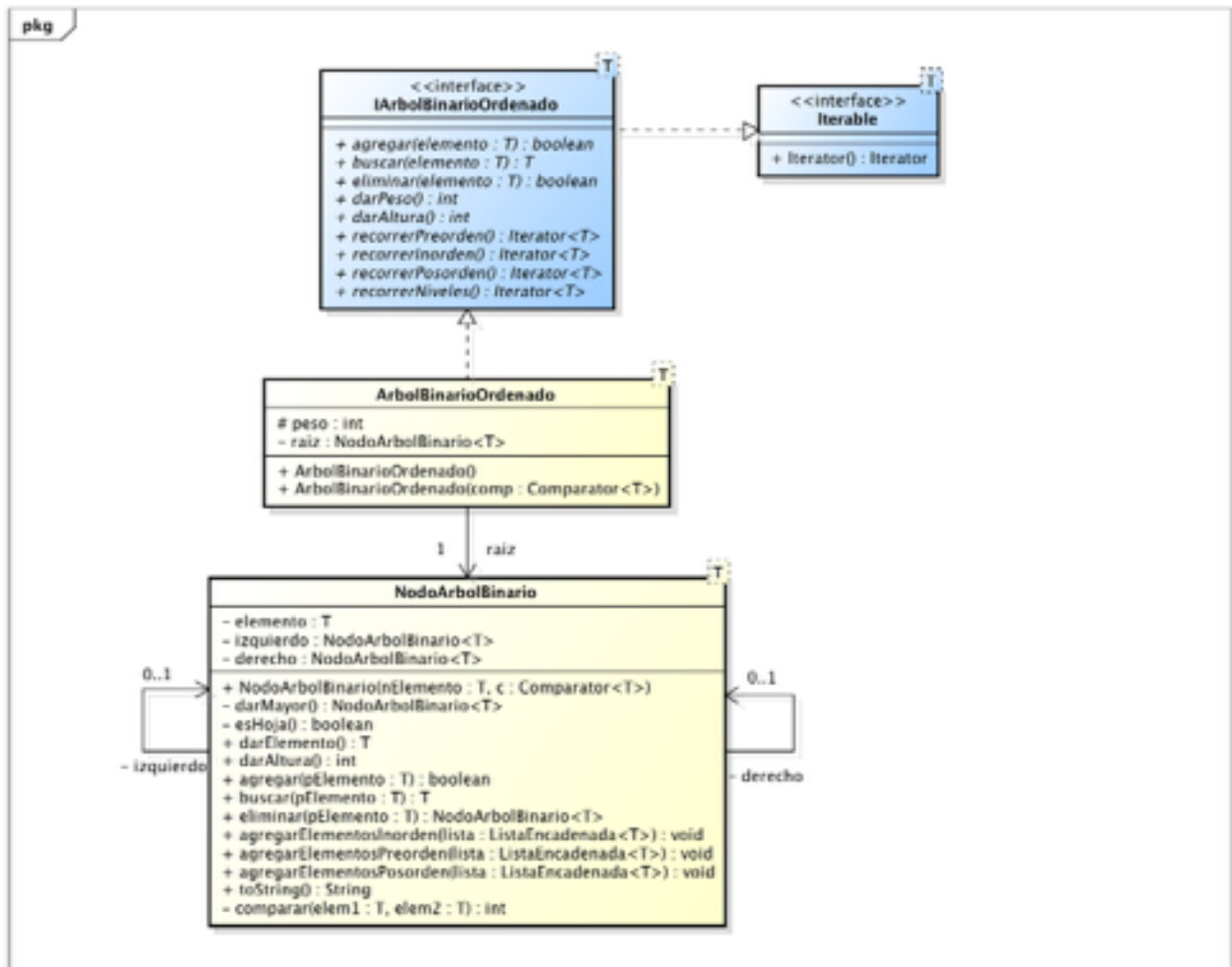
Al utilizar árboles AVL y árboles ordenados como estructuras de datos podemos garantizar una complejidad relativamente baja ($\log(n)$) para realizar búsquedas, inserciones y eliminaciones, garantizando un tiempo de complejidad mas bajo comparado con las listas encadenadas, las listas encadenadas ordenadas. Adicionalmente, la complejidad se mantiene constante y no varía, como con el caso de la TablaHashing donde las colisiones (que entran dentro de listas) pueden afectar la complejidad. Por último, las exploraciones que se realizan en el ejercicio tienen una forma de árbol ya que cada tag puede contener sus respectivos hijos, al igual que en las búsquedas.



6. Diseño final de la aplicación

Describe en detalle el diseño final de su aplicación a través de un diagrama UML, incluya atributos y métodos. Cuando se refiera a una estructura de datos no es necesario que incluya toda su definición dentro del diagrama. Basta con referenciar su interfaz. Basta con referenciar su interfaz.







7. *Justificación de decisiones de diseño del mundo*

Describe en detalle los aspectos que guiaron el diseño final de su aplicación.

En primer lugar, consideramos reducir el tiempo de procesamiento para realizar las búsquedas en la aplicación y para optimizar los procesos. Debido a estas razones, los árboles AVL constituyeron una parte fundamental del programa al garantizar un tiempo menor de ejecución garantizado para buscar los elementos deseados con una complejidad de $O(\log(n))$. Por otra parte, la librería de scraping, Jsoup, permite hacer búsquedas con selectores de CSS, por lo que definimos ciertos criterios de búsqueda para los elementos HTML que se desean buscar aprovechándonos de estos datos.



8. Análisis de complejidad de operaciones del mundo

Basado en los diseños de las estructuras de datos y en el mundo de la aplicación, defina la complejidad de cada una de las operaciones. Utilice la notación $O(f(n))$. Justifique por qué considera que esta es la mínima posible para el problema que está resolviendo.

| Caso | Complejidad | Justificación |
|------|--------------|--|
| R1 | $O(\log(n))$ | El usuario solo debe agregar al árbol AVL con los sitios un nuevo sitio. Corresponde a la complejidad del método agregar. |
| R2 | $O(\log(n))$ | Al especificar la profundidad de la exploración y debido a su carácter recursivo, la búsqueda con el Web-Crawler tiene una complejidad similar a agregar. (AVL) |
| R3 | $O(\log(n))$ | Para mostrar las estadísticas de un recurso particular se deben recorrer los recursos hasta buscar el seleccionado, por lo que la complejidad corresponde al buscar del AVL. |
| R4 | $O(n)$ | Para consultar el historial de operaciones realizadas se debe recorrer en inórden el árbol de operaciones realizadas, por lo que se recorren n elementos. |
| R5 | $O(\log(n))$ | La realización de la búsqueda según los criterios se realiza una búsqueda dentro del árbol de elementos resultantes. |
| R6 | $O(\log(n))$ | Tanto para agregar como eliminar, se recorre el árbol buscando/insertando elementos, por consiguiente, tienen ambos misma complejidad. |
| R7 | $O(\log(n))$ | Para adicionar o eliminar un recurso del árbol de recursos, la complejidad se mantiene para los métodos respectivos del árbol AVL (agregar/eliminar) |
| R8 | $O(n)$ | Debe recorrer al menos cada bit agregado para construir su representación comprimida. Al igual, depende de la cantidad de texto. |
| R9 | $O(k)$ | Para recuperar el conjunto de categorías del servidor, solo es necesario conectarse a éste y hacer el pedido respectivo. |
| R10 | $O(k)$ | Para visualizar el recurso selecciona o solo es necesario obtener el objeto y mostrarlo en el panel respectivo. |